Princeton Instruments

MEGAPLUS Central

Control Software for High Resolution Digital Cameras

User's Manual Version 6.0

4411-0117, Rev. B

For Use with *Mega*Plus Camera Controllers and the Following Camera Heads*:

EC 16000	ES 4020
EP 16000	ES 3200
EC 11000	ES 2093
EM 11000	ES 2020
EP 11000	ES 1603
ES 11000	

*The following models have been discontinued:
ES 2001 and ES 1602

©Copyright 2007-2009

Princeton Instruments, a division of Roper Scientific, Inc.
3660 Quakerbridge Rd
Trenton, NJ 08619
TEL: 800-874-9789 / 609-587-9797
FAX: 609-587-1970

All rights reserved. The information in this manual is for information purposes only and is subject to change without notice. Princeton Instruments makes no warranty of any kind with regards to the information contained in this manual, including but not limited to implied warranties of merchantability and fitness for a particular purpose. Princeton Instruments shall not be liable for errors contained herein nor for incidental or consequential damages from the furnishing of this information. No part of this manual may be copied, reproduced, recorded, transmitted or translated without the express written permission of Princeton Instruments, a division of Roper Scientific, Inc. ("Princeton Instruments").

Printed in the United States of America.

FireWire is a trademark of Apple Inc., registered in the U.S. and other countries.

GigE Vision is a trademark of the Automated Imaging Association

MEGAPLUS is a registered trademark of Redlake MASD, LLC LTD LIAB CO.

NI-IMAQdx is a trademark and National Instruments and NI are registered trademarks of National Instruments Corporation.

Windows and Windows NT are registered trademarks of Microsoft Corporation in the United States and/or other countries.

Table of Contents

1. Contact Information 1

2. Compliance Certifications 2
2.1 FCC Declaration 2
2.2 CE and Other Certifications 2

3. Precautions 3
3.1 DVI Output 3
3.2 Controller-to-Camera Head Cables 3
3.3 Laser Beams 3
3.4 Life Support Applications Policy 3
3.5 Non-critical Medical Applications 3
3.6 Shipping 3
3.7 Electromagnetic Fields 3

4. System Description 5
4.1 Features 6

5. Before You Begin 9
5.1 Overview 9
5.2 Package Contents 9
5.3 Recommended System Requirements 9
5.4 Operating System Requirements 9
5.5 Camera Control and Image Acquisition Interfaces 9
5.6 Select the Camera Head Configuration 11

6. Hardware Setup 13
6.1 Hardware Setup for the IEEE 1394a FireWire® Interface (multi-head controller only) 13
6.2 Hardware Setup for GigE Vision Interface (GigE Vision single-head controller only) 14
6.3 Hardware Setup for the CameraLink Interface 16
6.4 Camera Controller Front and Back Panel Connectors 17

7. Software Installation 21
7.1 Firewire 21
7.2 GigE Vision 27
7.3 CameraLink Frame Grabber Installation 32
7.4 Install the *MEGAPLUS* SDK (Optional) 34
7.5 How to Change Default Directories for Saving Files 35

8. *MEGAPLUS* Central Startup and Overview 37
8.1 System Controls Overview 37

9. Camera Settings 39
9.1 How to Set the Gain, Integration Time and Brightness (Offset) 39
9.2 Flat Field Normalization 40
9.3 Pixel Defect Concealment 57
9.4 White Balance 57
9.5 How to Display the Crosshairs for Color Cameras 58
9.6 How to Set the Trigger Parameters 59
9.7 How to Set the Strobe Parameters 59
9.8 Color Processing 60
9.9 Temperature Control 62

10. Save and Restore Camera Settings 63
10.1 How to Save Camera State Settings 64
10.2 How to Restore Camera Settings 64

11. Camera Head Configuration Settings 65
11.1 How to Select DVI Output 65
11.2 How to Change the Camera Configuration Settings 65
11.3 How to Reset the FireWire IEEE 1394a bus 65
11.4 Camera Control and Image Acquisition Configuration 66

12. Image Display and Acquisition 69
12.1 FireWire Image Acquisition 69
12.2 FireWire and *MEGAPLUS* Image Bit Depths and Image Formats 71
12.3 How to Configure the Transfer Format Using FireWire 71
12.4 GigE Vision Configuration and Output 72
12.5 CameraLink Configurations and Output 73
12.6 How to Set Multi-head Configurations with CameraLink 74
13. Operation Modes: Triggering or Continuous Video 77
13.1 Trigger Off; Free Run, Continuous Video 77
13.2 Trigger Modes 78
13.3 Mode 0; Edge Triggering (Asynchronous Reset with Programmable Integration) 79
13.4 Mode 1; Level Triggering (Asynchronous Reset with Pulse Width Controlled Integration) 84
13.5 Mode 4; Double Exposure Triggering 87
13.6 Mode 6; Periodic Interval Triggering (Self Trigger) 90
13.7 Mode 7; Overlap Triggering (Asynchronous Reset with Programmable Integration and Readout During Integration) 93
13.8 Retriggering 96
14. Serial Command Protocol 97
14.1 Introduction 97
14.2 Command Syntax 97
14.3 Camera Response to Commands 98
14.4 Nomenclature 99
14.5 Configuration Functions 99
14.6 Camera Head Related Functions 101
14.7 Controller Related Functions 111
14.8 Sensor-Related Functions 129
15. GigE Vision Features 131
15.1 Introduction 131
15.2 Camera Attributes 131
16. FireWire Camera Control 139
16.1 Standard Features Implementation 139
16.2 Advanced Features 139
16.3 IEEE 1394a FireWire Image Formatting 139
17. Connectors, Pin Outs & Cables 159
17.1 CameraLink MDR Connector 159
17.2 Power Connector Assignment 159
17.3 RS232 Serial DB9 Connector Assignment 160
17.4 Ethernet Cross-over Cable Pin Outs 160
18. Mechanical 161
18.1 Lens Mounts 161
18.2 LED Displays 161
18.3 ES 3200/1603/1602 Mechanical Shuttering 162
18.4 Camera Component Specifications 162
18.5 Sensor Quantum Efficiencies 173
18.6 Camera Component Mechanical Dimensions 181
19. Maintenance, Technical Support and Warranty 195
19.1 Maintenance 195
19.2 Technical Support 195
19.3 How to Create a Diagnostic Report 195
19.4 Warranty 196
20. Troubleshooting 197
20.1 Issues with NI-IMAQ Drivers 197
20.2 Issues with GigE Vision 197

21. Bit Windowing Overview 199

22. Color Space Correction 201

23. Updating *MEGAPLUS* Camera Controller Firmware via FTP 203
23.1 Items Needed for an Update 204
23.2 Overview 204
23.3 Performing the Update 205

Appendix A. GigE Vision -- Changing from One Camera to Another 209

Figures

Figure 4.1 *MEGAPLUS* System Configuration: (a) Single-Head Controllers, (b) Multi-Head Controller that Supports up to Four Camera Heads 5
Figure 5.1 CameraLink vs. FireWire Interface 11
Figure 6.1 Multi-Head Controller Front Panel 17
Figure 6.2 Single-Head Controller Front Panel 17
Figure 6.3 GigE Vision Single-Head Controller Front Panel 17
Figure 6.4 Multi-Head Controller Rear Panel 18
Figure 6.5 Single-Head Controller Rear Panel 18
Figure 6.6 GigE Vision Single-Head Controller Rear Panel 18
Figure 7.1 NI-Vision Acquisition Software Install Screen 21
Figure 7.2 NI Vision Features Dialog 22
Figure 7.3 Measurement and Automation Explorer Window 23
Figure 7.4 Driver Menu 23
Figure 7.5 MAX Window when *MEGAPLUS* Camera is Open 24
Figure 7.6 *MEGAPLUS* Software CD Menu 24
Figure 7.7 *MEGAPLUS* Central Setup Program 25
Figure 7.8 *MEGAPLUS* Camera Installer - First Screen 25
Figure 7.9 Specifying the Camera Data Interface 26
Figure 7.10 *MEGAPLUS* Central Main Screen 27
Figure 7.11 NI-Vision Acquisition Software Install Screen 28
Figure 7.12 NI Vision Features Dialog 29
Figure 7.13 *MEGAPLUS* Central Gigabit Ethernet Connection 32
Figure 7.14 *MEGAPLUS* Central Camera Control Window 32
Figure 7.15 *MEGAPLUS* Central Camera Control Window 33
Figure 7.16 MEGAPLUS Lib SDK Installation 34
Figure 7.17 Changing Default Directories Dialog Box 35
Figure 8.1 *MEGAPLUS* Central Main Window 37
Figure 8.2 *MEGAPLUS* Central Main Window and Serial Debug Window 38
Figure 9.1 Camera Head Control Window- Basic Tab 39
Figure 9.2 Image of an Evenly Illuminated Flat Field with Shading 40
Figure 9.3 Pixel Values at the Horizontal Center of the Image 40
Figure 9.4 Normalization Calibration Window 42
Figure 9.5 Get Dark Field Image Dialog Box 43
Figure 9.6 Get Flat Field Image Dialog Box 44
Figure 9.7 Get Dark Field Image Dialog Box 45
Figure 9.8 Get Flat Field Image Dialog Box 45
Figure 9.9 Calculate Correction Dialog Box 46
Figure 9.10 Download Normalization Tables Dialog Box 47
Figure 9.11 Apply Normalization Dialog Box 48
Figure 9.12 Histogram and Profile display (Normalization and Calibration window) 48
Figure 9.13 Normalization Calibration Window 49
Figure 9.14 Get Dark Field Image Dialog Box 50
Figure 9.15 Get Flat Field Image Dialog Box 51
Figure 9.16 Histogram and Profile Display (Normalization Calibration Window) 52
Figure 9.17 Calculate Correction Dialog Box 53
Figure 9.18 Download Normalization Tables Dialog Box 53
Figure 9.19 Apply Normalization Dialog Box 54
Figure 9.20 Camera Head Control Window- Color Tab 58

Figure 9.21 Trigger and Strobe Settings 60
Figure 9.22 Controller Control Gamma and Color Space Conversion 61
Figure 11.1 Control Panel - System Tab 66
Figure 11.2 System Camera List - Camera Control 67
Figure 12.1 Image Display 70
Figure 12.2 ROI Dialog Box 71
Figure 12.3 Image Display 73
Figure 12.4 Output Data Tab - CameraLink Dual Base, Alt Tap 75
Figure 13.1 Free Run Mode - No External Trigger 77
Figure 13.2 Basic Edge Trigger Mode – User-programmable Integration 79
Figure 13.3 Triggering with No Strobe Delay for Interline Devices 80
Figure 13.4 Triggering with Strobe Delay for Interline Devices 80
Figure 13.5 Mode 0 Timing Diagram for Full Frame Sensors 83
Figure 13.6 Integrate and Dump - Level Controlled Trigger Mode 84
Figure 13.7 Triggering with No Strobe Delay 84
Figure 13.8 Level Mode Triggering for Full Frame Sensors with a Strobe Delay 87
Figure 13.9 Double Exposure Timing 88
Figure 13.10 Periodic Interval Trigger Timing 90
Figure 13.11 Overlap Trigger Mode Timing 93
Figure 13.12 Retriggering of a Camera during Integration 96
Figure 13.13 Retriggering of a Camera during Transfer or Readout 96
Figure 16.1 Format 7 ROI Capability allows Capture of Partial Images 142
Figure 18.1 KAI-16000 Mono Quantum Efficiency 173
Figure 18.2 KAI-16000 Color Quantum Efficiency 173
Figure 18.3 KAI-11002 Mono Quantum Efficiency 174
Figure 18.4 KAI-11002 Color Quantum Efficiency 174
Figure 18.5 KAI-4021 Mono Quantum Efficiency 175
Figure 18.6 KAI-4021 Color Quantum Efficiency 175
Figure 18.7 KAF 3200 ME Mono Quantum Efficiency 176
Figure 18.8 KAI-2093 Mono Quantum Efficiency 176
Figure 18.9 KAI-2093 Color Quantum Efficiency 177
Figure 18.10 KAI-2020 Mono Quantum Efficiency 177
Figure 18.11 KAI-2020 Color Quantum Efficiency 178
Figure 18.12 KAI-2001 Mono Quantum Efficiency 178
Figure 18.13 KAI-2001 Color Quantum Efficiency 179
Figure 18.14 KAF 1602 Quantum Efficiency 179
Figure 18.15 KAF 1603 Quantum Efficiency 180
Figure 18.16 EC 16000/11000 Mechanical Dimensions Front View with F-mount 181
Figure 18.17 EC 16000 & EC 11000 Mechanical Dimensions Side View with F-mount 181
Figure 18.18 EC 16000/11000 Mechanical Dimensions Bottom View with F-mount 182
Figure 18.19 EM 11000 Mechanical Dimensions Front View with F-mount 182
Figure 18.20 EM 11000 Mechanical Dimensions Side View with F-mount 183
Figure 18.21 EM 11000 Mechanical Dimensions Top View with F-mount 183
Figure 18.22 EP 16000/11000 Mechanical Dimensions Front View with F-mount 184
Figure 18.23 EP 16000/11000 Mechanical Dimensions Side View with F-mount 185
Figure 18.24 EP 16000/11000 Mechanical Dimensions Top View with F-mount 185
Figure 18.25 ES 11000 Mechanical Dimensions Front View with F-mount and Bottom Plug 186
Figure 18.26 ES 11000 Mechanical Dimensions Side View with F-mount and Bottom Plug 186
Figure 18.27 ES 11000 Mechanical Dimensions Bottom View with F-mount and Bottom Plug 187
Figure 18.28 ES 4020/2093/2020/2001 Mechanical Dimensions Side View with F-mount 187
Figure 18.29 ES 4020/2093/2020/2001 Mechanical Dimensions Front View with F-mount 188
Figure 18.30 ES 4020/2093/2020/2001 Mechanical Dimensions Top View with F-mount 188
Figure 18.31 ES 3200/1603/1602 Mechanical Dimensions with F-mount 189
Figure 18.32 ES 3200/1603/1602 Mechanical Dimensions with C-mount 189
Figure 18.33 Multi-Head Controller Rear View, Rotated 90° 190
Figure 18.34 Multi-Head Controller Front View, Rotated 90° 190
Figure 18.35 Multi-Head Controller Bottom View, Rotated 90° 191
Figure 18.36 Single-Head Controller Top View 192
Figure 18.37 Single-Head Controller Side View 193

Figure 18.38 Single-Head Controller Rear View 193
Figure 18.39 GigE Vision Single-Head Controller 194
Figure 21.1 Bit Window Example 199
Figure 22.1 CIE 1931 Color Curve 201
Figure 22.2 KAI-11002 Color Quantum Efficiency 202
Figure 22.3 *MEGAPLUS* Color Correction Model 202
Figure 23.1 Confirming the Connection using "Ping" 206
Figure 23.2 An Example of an Update Session 207
Figure 23.3 Resetting TCP/IP Properties to Pre-update Values 208

Tables

Table 4.1 Camera Models & Performance Specifications 7
Table 4.2 Controller Models & Performance Specifications 8
Table 9.1 Trigger Mode Selection 59
Table 10.1 Saved Camera State Settings 63
Table 13.1 Minimum Integration Times for Free Run Mode 78
Table 13.2 Sensor/Sensor Frequency/Tcp 81
Table 13.3 Mode 0 Minimum Integration Times 81
Table 13.4 Sensor/Sensor Frequency/Txfr 82
Table 13.5 Sensor/Sensor Frequency/ Single Tap/Dual Tap/Tread Interline Sensors 82
Table 13.6 Sensor Clock Rate/Tread for Full Frame Sensors 83
Table 13.7 Sensor/Sensor Frequency/Tcp 85
Table 13.8 Sensor/Sensor Frequency/Txfr 85
Table 13.9 Sensor/Sensor Frequency/ Single Tap/Dual Tap/Tread 86
Table 13.10 Strobe Delay for Double Exposure Mode 88
Table 13.11 Transfer Time for Double Exposure Mode 89
Table 13.12 Maximum Trigger Rate for Double Exposure 89
Table 13.13 Sensor/Pixel Clock Rate/Ts 91
Table 13.14 Sensor/Pixel Clock Rate/Txfr 91
Table 13.15 Sensor/Taps/Sensor Frequency/Tread 92
Table 13.16 Sensor/Pixel Clock Rate/Ts 94
Table 13.17 Sensor/Pixel Clock Rate/Txfr 94
Table 13.18 Sensor/Taps/Sensor Frequency/Tread 95
Table 16.1 Video Modes and Image Output Formats 140
Table 16.2 IEE-1394 FireWire Data Packets Settings 141
Table 16.3 Gain Count Values-Red 156
Table 17.1 CameraLink 26 Pin MDR Connector Pin Assignments 159
Table 17.2 RS232 Serial DB9 Connector Assignment 160
Table 17.3 Ethernet Cross-over Pin Outs 160
Table 18.1 Lens Mounts 161
Table 18.2 EC 16000/11000 LED Display 161
Table 18.3 EP 16000/11000 LED Display 161
Table 18.4 EM 11000 LED Display 161
Table 18.5 EC/EP 16000 Camera Head Specifications 163
Table 18.6 EC/EM/EP/ES 11000 Camera Head Specifications 165
Table 18.7 ES 4020 Camera Head Specifications 165
Table 18.8 ES 3200 Camera Head Specifications 166
Table 18.9 ES 2093 Camera Head Specifications 167
Table 18.10 ES 2020 Camera Specifications 168
Table 18.11 ES 2001 Camera Head Specifications 169
Table 18.12 ES 1602/1603 Camera Head Specifications 170
Table 18.13 Multi-Head Controller Specifications 171
Table 18.14 Single-Head Controller Specifications 171
Table 18.15 GigE Vision Single-Head Controller Specifications 172

This page intentionally left blank.

1. Contact Information

Americas:

Princeton Instruments
3660 Quakerbridge Road
Trenton, NJ 08619

Toll Free in USA: +1 877.474.2286
Tel: +1 609.587.9797
Fax: +1 609.587.1970
Email: moreinfo@princetoninstruments.com
Website: http://www.princetoninstruments.com

European Union:

Roper Scientific, GmbH
Rosenheimer Landstr. 87
D-85521 Ottobrunn Germany

Tel: + 49 (0) 89-660 779 3
Fax: + 49 (0) 89-660 779 50
Email: contactus@roperscientific.de

China:

PI - China - Regional Office
Room 12A15 Yingu Building
No. 9 Beishihuanxi Road
Haidian District
Beijing, China 100080

Tel: +86.10.8252.5122
Fax: +86.10.8252.5327
Mobile: +86135 0122 8135
Email: tguo@princetoninstruments.com

Singapore and Taiwan:

PI - Asia - Pacific Regional Office
10 Eunos Road 8
Singapore Post Centre #12-06
Singapore 408600

Office Tel: +65 6408 6240
Patrick Cullen Mobile: +65 9669 1247
Fax: +65 6293 3307
Email: pcullen@princetoninstruments.com

Japan:

PI - Japan - Regional Office
Nippon Roper, K. K.
Sakurai Building
2-8-19 Fukagawa, Kotu-Ku
Tokyo 135-0033, JAPAN

Tel: +81.3.5639.2741
Fax: +81.3.5639.2775
Email: sales@roper.co.jp

To find local representatives for specific regions, please refer to www.princetoninstruments.com

Send comments regarding this manual to moreinfo@princetoninstruments.com

2. Compliance Certifications

2.1 FCC Declaration

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna
- Increase the separation between the equipment and receiver
- Connect the equipment into an outlet on a circuit different from the one the receiver is connected
- Consult the dealer or an experienced radio/TV technician for help

2.2 CE and Other Certifications

MEGAPLUS® Camera Heads and Camera Controllers have been tested and are fully compliant with the following:

European Union and Australia/New Zealand
EN61326 (1997 wA1: 98 & A2: 01) Class B,
CISPR, Class B

Japan
VCCI (April 2000) Class B

Canada
ICES-003 Class B (ANSI C63.4 1992)

3. Precautions

3.1 DVI Output

DO NOT use the DVI output option for extended periods on Camera Controllers that are not equipped with an integrated cooling fan. Using the DVI output option generates significant heat inside the controller.

3.2 Controller-to-Camera Head Cables

DO NOT HOT PLUG THESE CABLES. Turn the Camera Controller power off when installing or removing a Camera Head.

DO NOT bend cables more sharply than a 3 inch radius curve or more than 60 degrees.

NOTE: The controllers and camera heads shipped since October 2007 have "polarizing" brackets to prevent accidentally connecting head cables upside down.

3.3 Laser Beams

A laser beam focused on the sensor, either directly or by reflection, can cause permanent damage to the sensor. Any laser powerful enough to produce localized heating at the surface of the sensor will cause damage, even if the camera power is off. A sensor damaged by laser light is NOT covered by the warranty.

3.4 Life Support Applications Policy

MEGAPLUS® cameras are not authorized for and should not be used with life support systems without the specific written consent from Princeton Instruments.

3.5 Non-critical Medical Applications

MEGAPLUS cameras must be grounded while operating.

3.6 Shipping

When shipping, use a carton that protects the camera from shock and moisture, similar to the carton in which the unit was originally delivered. Do not ship the equipment in a cargo area where the temperature will drop below -25º C or exceed 70º C.

3.7 Electromagnetic Fields

Do not operate the camera in the vicinity of strong electromagnetic fields. Avoid electrostatic discharging.

This page intentionally left blank.

4. System Description

The *MEGAPLUS*® design consists of up to four camera heads and a controller, which is seamlessly integrated using Princeton Instruments' proprietary high data rate Rocket I/O digital interface. All *MEGAPLUS* cameras require a Camera Controller and include the easy-to-use *MEGAPLUS* Central software. This software features robust preprocessing algorithms such as gamma correction, real-time color space conversion, and more.

- Very low noise, high dynamic range sensor electronics with 12-bit digitalization
- Pixel-by-pixel gain and offset correction
- FireWire® IEEE 1394a (multi-head controller only), GigE Vision (GigE Vision-compliant single-head controller only), and CameraLink digital image output
- Simultaneous acquisition with up to four Camera Heads
- Advanced Bayer color filter array de-multiplexing
- Color space conversion
- Multiple trigger modes including double triggering and internal triggering
- Field upgradeable firmware
- Low dark noise and thermal drift with the EC 11000 and EC 16000 actively cooled cameras

The cameras are available either with a Single-Head Controller or a Multi-Head Controller that can support up to four camera heads. The Camera Controller houses the camera calibration and imaging electronics, and connects to the host PC via FireWire, Gigabit Ethernet, or CameraLink. By separating the camera electronics and image handling software from the sensors, the standard *MEGAPLUS* cameras are able to maintain cooler sensor temperatures and achieve lower noise and higher dynamic range than comparable cameras.

Figure 4.1 *MEGAPLUS* System Configuration: (a) Single-Head Controllers, (b) Multi-Head Controller that Supports up to Four Camera Heads

NOTE: The information provided in this manual applies to system configuration based on Single-Head Controller **and** Multi-Head Controller, unless otherwise indicated.

4.1 Features

The following features are all controlled by the *MEGAPLUS* Central Camera Control Software. Camera features can also be controlled via the MEGAPLUS Lib SDK, direct serial communications control, the serial link embedded in the CameraLink connection, GigE Vision features, or via 1394 camera registers. Availability of some features depends on the specific firmware configuration of the camera.

Bayer Demux: In-Camera Color Filter Array (CFA) de-multiplexing using a proprietary algorithm that minimizes color aliasing and maximizes the resolution of the three color planes.

Bit Depth/Bit Window Selection: Enables users to select the bit-depth of data output from the camera. All internal data is 12 bits per image. When bit depths less than 12 bits are selected, the least significant bit can be specified in order to select which of the 12 available bits is output, creating a "bit window."

Brightness (Offset): Controls the black level or offset of the image by specifying a digital number to be added to all image pixel data. This value can be written to and read from the camera. For multi-head or multi-sensor configurations, brightness can be specified for each sensor.

Camera Settings Read/Write: Allows the camera to save its current operating parameters to internal, non-volatile memory within the Camera Controller. A set number is used as an identifier for the record. The record of saved operating parameters can then be applied to the camera. Multiple sets of camera parameters can be stored and retrieved.

Crosshair Display (Non-GigE Vision configurations only): Superimposes a marker in the image indicating the absolute digital center of the image. Reticle or crosshair display can be turned on and off on a per-sensor basis.

Color Space Conversion and Correction (Non-GigE Vision configurations only): Allows programmatic control of an internal color space conversion engine in the Camera Controller. The color space engine consists of 3 input Look up Tables (LUTS), a 3x3 matrix with user-defined coefficients, and 3 output LUTS.

Defective Pixel Correction: Allows the identification and concealment of defective pixels. A list of defective pixels can be stored in the camera and concealment of these pixels turned on and off.

Gain: Controls the gain applied at the sensor. Values are specified in dB. The value can be written to or read from the camera. For multi-head or multi-sensor configurations, gain can be specified for each sensor. Gain values can range from 0 dB to 34 dB with 1024 levels (10 bits).

Gamma: Defines a relationship between the incoming light level and the output digital number. Specifying a gamma value is useful to stretch the dark areas of an image making the information better suited for viewing with the eye. In addition, gamma lookup tables can be defined by the user and downloaded to the camera as a part of the color space conversion engine.

Integration Time: Controls the electronic integration period of the camera. It is identified as shutter under IIDC. Values are specified in milliseconds. The value can be written to or read from the camera.

Output Multiplexing (4-head mono configurations only): Controls the mapping of image data to CameraLink output ports. The camera may have several sets of image data that are derived directly from the sensors or from the Bayer demux process. Any available image data stream can be assigned to an output tap.

Mechanical Shutter Control (Full Frame Sensors only): Provides controls to activate the shutter.

Strobe Output: The strobe output from the camera provides the trigger signal as an output so that it can be used to control strobed illumination sources or other synchronous events. This function specifies whether the strobe polarity is negative or positive.

Trigger: Uses external signals to control when an image is acquired. The camera provides a variety of trigger modes. Trigger logic can be specified as negative or positive. Trigger signals can be sourced from the connector on the rear panel of the camera or when available, through the frame grabber cable. Software triggers are also supported. The camera provides a Strobe Out signal that relays the trigger signal out of the camera for use as a control strobe.

White Balance: Adjusts the relative intensity of the blue and red values in relation to green until all three colors read approximately the same digital number. The assumption is made that the camera is imaging a white color when the balance values are measured. The white balance function can be operated in manual or semi-automatic mode. In manual mode, blue and red adjustment values can be written to and read from the camera. In automatic mode, the camera performs an iterative process of acquiring images and adjusting sensor parameters until a white balance condition is achieved.

Cooling Performance: The EC 11000 and EC 16000 are actively cooled high-resolution 11 and 16 megapixel digital cameras, respectively. They use Peltier cooling to significantly reduce both dark noise and thermal drift. They are designed to cool the sensor below the ambient temperature.

NOTE: *MEGAPLUS* Camera Controller is required for all camera heads listed below.

Camera Type	Resolution	Sensor Type	Pixel Size	Frame Rate at Full Resolution (CameraLink)	Head Variations Available
EP/EC 16000	4872 x 3248	Interline	7.4 µm	3.2	EP-Passively Cooled EC-Actively Cooled
ES/EP/EC/EM 11000	4008 x 2672	Interline	9 µm	4.6	ES-Uncooled EP-Passively Cooled EC-Actively Cooled EM-Miniature Head
ES 4020	2048 x 2048	Interline	7.4 µm	15	ES-Uncooled
ES 3200	2184 x 1472	Full frame	6.8 µm	2.9**	ES-Uncooled
ES 2093	1920 x 1080	Interline	7.4 µm	30	ES-Uncooled
ES 2020	1600 x 1200	Interline	7.4 µm	30	ES-Uncooled
ES 1602*/1603	1536 x 1024	Full frame	9 µm	6.4**	ES-Uncooled

* ES 1602 has been discontinued.

**Does not include shutter integration time

Table 4.1 Camera Models & Performance Specifications

Feature	Single-Head Controller	GigE Vision Single-Head Controller	Multi-Head Controller
Number of Heads Supported	1	1	Up to 4
CameraLink	Yes (Base)	No	Yes (Base, Medium, Dual Base)
GigE Vision	No	Yes	No
FireWire	No	No	Yes
RS 232	Yes	Yes	Yes
DVI Video Output	No	No	Yes
Trigger In/Out	Yes	Yes	Yes
Field Updateable Firmware	Yes	Yes	Yes
Dimensions (LxDxH)	158.8 x 152.0 x 38.1 mm	158.8 x 152.0 x 54.4 mm	160.7 x 158.3 x 50.8 mm
Weight	992 g	1124 g	1140 g

Table 4.2 Controller Models & Performance Specifications

5. Before You Begin

5.1 Overview

- Check your package contents to ensure you have all of the necessary components.
- Select the data interface and install the required hardware and software.
- Decide which camera configuration best suites your needs.
- Determine if you will be developing in-house applications.

5.2 Package Contents

The total package you receive will depend on the number of Camera Heads and components you ordered. Please check to make sure that at a minimum, all of the items listed are included in your package.

5.2.1 Camera Controller Package

- Camera Controller with one or four head connectors
- NI-Vision Acquisition Software
  - NI-IMAQ IEEE 1394a Drivers Version. 2.0 or higher (multi-head controller only)
    (download upgrades from the NI website at www.ni.com)
  - NI-IMAQdx™ Drivers Version 3.2.1 or higher (GigE Vision single head controller only)
- CD with User's Manual, Quick Start Guide, *MEGAPLUS* Central Control Software
- Power Supply

5.2.2 Camera Head and Cables

Camera Heads and cables are sold separately.

- Camera Head(s)
- Camera Head cable(s)

5.3 Recommended System Requirements

The *MEGAPLUS*'s flexibility in applications and interface configurations produces a wide range of system requirements. At a minimum, follow the requirements of the CameraLink Frame grabber, FireWire interface, or Gigabit Ethernet interface manufacturer.

5.4 Operating System Requirements

Microsoft Windows™ operating system: Windows XP (SP2 or later) or Windows 2000 Pro (SP2 or later).

5.5 Camera Control and Image Acquisition Interfaces

The *MEGAPLUS* imaging system offers a choice of camera control and image acquisition interfaces. Before beginning the installation, decide which interface option best suits your requirements.

MEGAPLUS Central auto-configures based on whether FireWire control, GigE Vision control, or Serial (CameraLink or RS-232) control is selected. Since some *MEGAPLUS* functions are interface specific, buttons and pull-down menus are dimmed when they are not active.

The Camera Control panel has a pull-down menu at the top of the panel that is used to select the output configuration for individual Camera Heads. For more flexibility, Custom

can be selected which enables configuration of individual ports for CameraLink when either the DB9 or CameraLink serial ports are selected for communications.

Interfaces that handle both Camera Control and Image Acquisition:

- IEEE 1394a FireWire (multi-head controller only)
  (OHCI compliant supports 400 Maps or S400 transfers)
- GigE Vision (GigE single-head controller only)
- CameraLink Frame Grabber

Interface for Camera Control Only:

- Standard RS-232

5.5.1 IEEE 1394a FireWire (multi-head controller only)

For those installations with less demanding bandwidth requirements, an IEEE 1394a FireWire interface is provided for both camera control and image acquisition. The camera meets the Instrumentation and Industrial Digital Camera Working Group's standard (IIDC Version 1.3) for camera control. This option eliminates the need for a frame grabber and enables the use of the camera in a networked configuration on a 1394 bus. Some computers have an integrated 1394 port. If your system does not, you will need to add 1394 interface hardware. Follow the vendor's instructions for installation.

5.5.2 GigE Vision (single-head GigE controller only)

GigE Vision is a camera interface standard developed based on Gigabit Ethernet communication protocol. It allows maximum data bandwidth of 125MBytes/sec over up to 100 meters of Cat 5e/6 cables. The GigE Vision standard is based on Gigabit Ethernet (GigE) protocol. However, it is customized for machine vision applications with a goal to offer more reliable image data transmission and a uniform camera control standard. All MEGAPLUS cameras are supported under GigE Vision. Although it is possible to acquire data from any PC with a GigE port, it is highly recommended that you use a dedicated Gigabit Ethernet Network Interface Controller (NIC) card based on Intel Pro 1000 chipset.

GigE Vision compliant acquisition software, such as NI's IMAQdx Vision, that supports the GigE Vision standard is required. Compliant software will also install the required drivers.

5.5.3 CameraLink Frame Grabbers

A standard CameraLink interface is available for interfacing to industry-standard frame grabbers. The CameraLink frame grabber interface provides a high bandwidth connection for image acquisition as well as an interface for camera control. The CameraLink interface can be used in Base, Medium or Dual output formats. In addition, *MEGAPLUS* also supports Alternating Tap output in the Base (Single-Head Controller) and Dual Base (Multi-Head Controller) configurations.

Alternating tap divides the output image pixel stream into two parallel streams, one consisting of the odd pixels and the other consisting of the even numbered pixels. This allows users to operate camera heads at their maximum frame rate at half the pixel clock rate required for single tap output. Please contact a Princeton Instruments representative for additional information regarding frame grabber support. The list of supported frame grabbers is constantly growing so be sure to check with Princeton Instruments for an update.

NOTE: Selecting an interface does not commit you to this interface method permanently. The interface is easily changed at any time. Refer to the Camera Control Configuration Section for step-by-step instructions. Specification of an interface during the installation process is necessary in order to configure the camera control software.

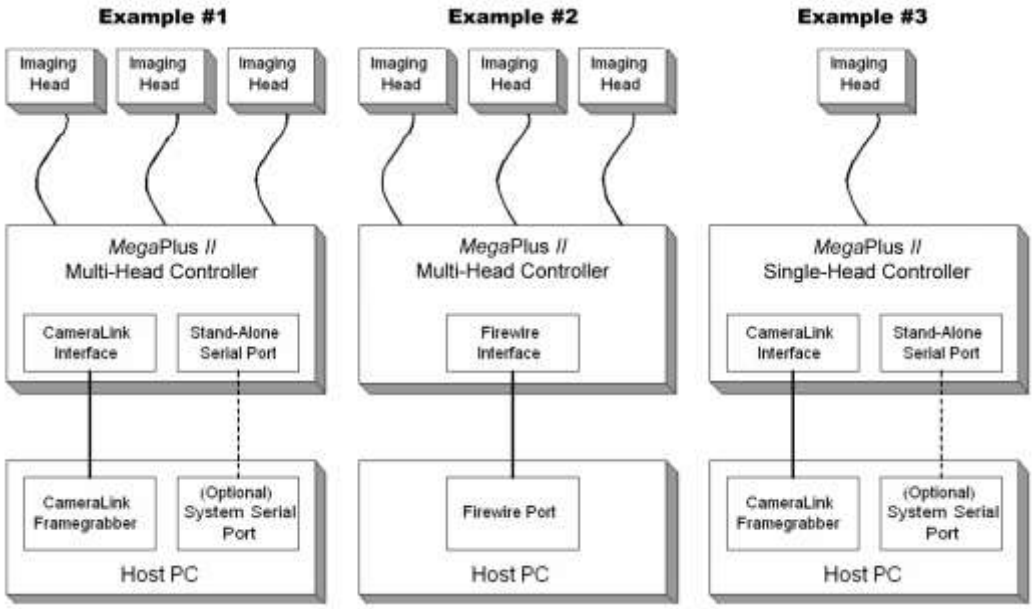


Figure 5.1 CameraLink vs. FireWire Interface
(Multi-head Controllers Support up to Four Heads)
(FireWire Interface is Available for Multi-Head Controllers Only)

Related Information:
See "Camera Controller Front and Back Panel Connectors" on page 17.

5.6 Select the Camera Head Configuration

Applies to multi-head controllers only.

MEGAPLUS cameras use a Field Programmable Gate Array (FPGA) for very high speed real time image processing. Currently, there are three sets of FPGA code tailored to provide three different feature sets. The FPGA configuration can be selected using the System tab in the Controller Control pane of the *MEGAPLUS* Central software

5.6.1 Configuration 1 - Dual Head, Advanced Color

Dual Head color mode supports EC/EP 16000, EC/EP/ES/EM 11000, ES 4020, ES 3200, ES 2093, and ES 2020. Provides advanced Bayer de-multiplexing using a proprietary algorithm for high resolution, minimal color aliasing separation of the color planes. Provides color space conversion/correction using of the color space engine. Supports all trigger modes except double exposure.

5.6.2 Configuration 2 - Four Head Mono

Four head mono mode supports EC/EP/ES/EM 11000, ES 4020, ES 3200, ES 2093, ES 2020, ES 2001*, and ES 1603/1602* (11000 is supported only up to 3 heads per controller). Supports color heads but they are treated as mono so that the raw Bayer data is output on the CameraLink interface. Supports all trigger modes including double exposure. It provides simultaneous output from the Camera Heads to user configurable CameraLink channels.

5.6.3 Configuration 3 - Dual Head Mono

Dual head mono mode supports up to two EC/EP 16000 and EC/EP/ES/EM 11000.

5.6.4 Developing In-House Applications

If you would like to write your own applications that interface to your *MEGAPLUS* camera, the MEGAPLUS System Developer's Kit (SDK) provides a consistent programmatic interface to the camera features via the MEGAPLUSLib Dynamic Link Library (DLL). A developer's reference manual is also available on the software installation CD or from the FTP site.

MEGAPLUS Central calls the MEGAPLUS Camera Control DLL for access to camera control features. This DLL is available in the *MEGAPLUS* System Developer's Kit (SDK) and provides a software interface for use in your own application development. The DLL is a standard C callable library accessible from a variety of programming environments.

Related information:
See "Serial Command Protocol" on page 97.

6. Hardware Setup

6.1 Hardware Setup for the IEEE 1394a FireWire® Interface (multi-head controller only)

6.1.1 How to Install your IEEE 1394a Interface Hardware

CAUTION! Make sure the MEGAPLUS Camera Controller power switch is turned to the OFF position before connecting the cables!

Some computers have an integrated 1394 port. If your system does not, you will need to add 1394 interface hardware. You will need the following items:

- IEEE 1394a FireWire interface OHCI Compliant host controller that supports 400 Mbps or S400 transfer.
- 6-pin to 4-pin FireWire cable that is 14 feet or shorter.

Follow the vendor's instructions for installation.

Before proceeding, make sure that your 1394 controller interface is installed and operating properly. You can verify that the system has properly identified the controller by checking in the Microsoft Windows Device Manager. It must be configured to use NI-IMAQ 1394.

NOTE: The 1394 connector on the Controller is a 4-pin connector. You must use a 4-pin to 6-pin or 4-pin to 4-pin cable as appropriate.

Related information:
See "Camera Control and Image Acquisition Interfaces" on page 9.
See "Overview" on page 21.

6.1.2 How to Connect the Camera Head, PC, and Camera Controller

Follow the steps below to connect the camera system to a PC for operation with an IEEE 1394a interface.

1. Connect the cable from the Camera Head to the camera port labeled Remote Head #1 on the Camera Controller. Additional Camera Heads can be connected to consecutive ports as needed.
2. Connect an IEEE 1394a cable from the IEEE 1394a connector on back panel of the Camera Controller to the host computer.
3. Connect the Camera Controller to the power supply.
4. If using an EC 11000 or EC 16000, connect the Camera to the auxillary power supply. The Thermal Electronic Cooler (TEC) and Fan will not work unless the Camera is plugged into a separate power source.

   NOTE: Do not combine EC 11000 Camera Heads with other Camera models on the same Camera Controller.

6.2 Hardware Setup for GigE Vision Interface (GigE Vision single-head controller only)

6.2.1 Recommended Computer Configuration

Host Computer: For enhanced performance, a fast hard drive (10,000 rpm) and 2GB RAM is recommended.

Operating System: Windows 2000, Windows XP (32-bit, SP3 or later) or Vista (32-bit)

Computer Interface Components:

- Cat5e/Cat 6 Ethernet Cable
- Dedicated GigE Interface card with Intel Pro/1000 chipset

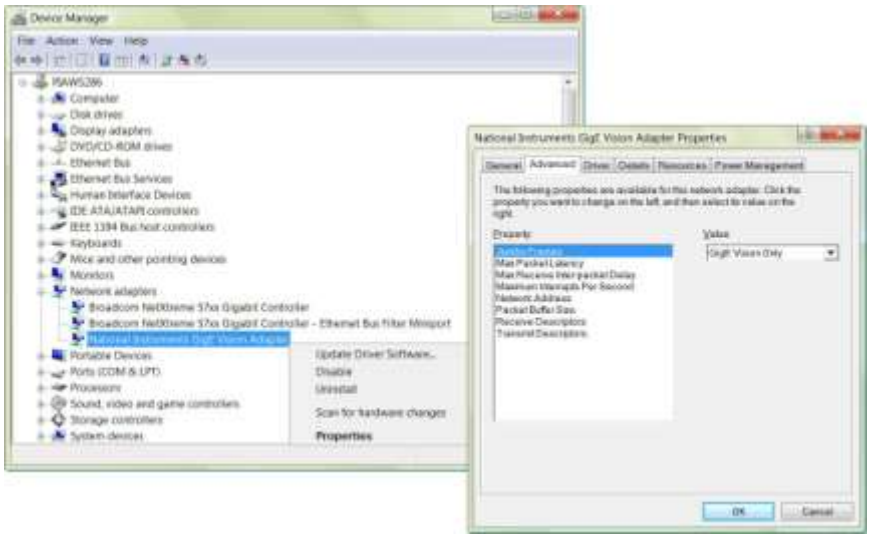
6.2.2 How to Install the GigE Interface Card

CAUTION! Make sure the MEGAPLUS Camera Controller power switch is turned to the OFF position before connecting the cables!

If your GigE interface card is not yet installed, you must install it and the manufacturer's drivers. Princeton Instruments recommends that you use a dedicated Gigabit Ethernet Network Interface Controller (NIC) card based on Intel Pro/1000 chipset.

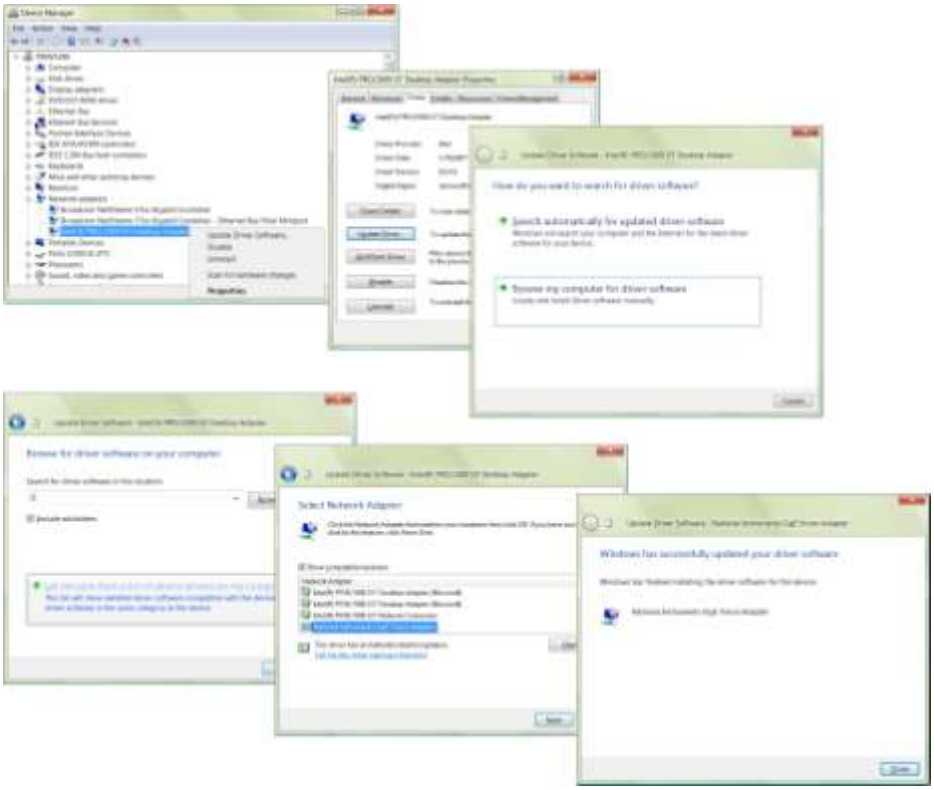
To Install a New GigE NIC card in your PC:

1. Following the computer manufacturer's instructions, install the card in your PC, then install the manufacturer's drivers. You can do so with the manufacturer's driver installer or your existing Driver Installation Tool.
2. Reboot your PC (do not skip this step!).
3. Access the Windows Device Manager and verify that the National Instruments GigE Vision Adapter has been installed. If the Intel Pro 1000 is listed instead, you will need to update the driver (see ***"To Update Intel PRO/1000 Driver to the National Instruments GigE Vision Adapter"***). Otherwise, right-mouse click on the National Instruments GigE Vision Adapter.
4. Click on **Properties**, select the **Advanced** tab, and verify that **Jumbo Frames** are available.



To Update Intel PRO/1000 Adapter to the National Instruments GigE Vision Adapter:

1. If the Intel PRO/1000 National Instruments Adapter is listed under Network Adapters and the National Instruments GigE Vision Adapter is not, you will need to update the driver.
2. Right-mouse click on the Intel Adapter, click on **Properties**, click on **Update Driver**…, and then select "**Browse my computer…"**
3. Click on **"Let me pick from a list…"**, select **National Instruments GigE Vision Adapter** from the list, click on **Next**, and wait for the driver update to conclude.
4. Close the window.
5. It may take up to 60 seconds for the IP address to be resolved.



6.2.3 How to Connect the Camera, PC, and Camera Controller

1. Connect the cable from the Camera to the port labeled Remote Head on the GigE Vision single-head Camera Controller.
2. Connect a Cat5e/6 Ethernet cable from the GigE Vision connector on the back panel of the Camera Controller to the Ethernet port of the dedicated GigE board in your PC.
3. Connect the Camera Controller to the power supply.
4. If using an EC 11000 or EC 16000, connect the Camera Head to the auxillary power supply. The Thermal Electronic Cooler (TEC) and Fan will not work unless the Camera is plugged into a separate power source.

6.3 Hardware Setup for the CameraLink Interface

6.3.1 How to Connect the Camera, PC, and Camera Controller

CAUTION! Make sure the MEGAPLUS Camera Controller power switch is turned to the OFF position before connecting the cables!

Before proceeding, make sure that your CameraLink frame grabber hardware and software are installed properly.

For operation with a CameraLink frame grabber, connect the camera and cables as follows:

1. Connect the cable from the Camera to the port labeled Remote Head #1 on the Camera Controller. Additional Camera Heads can be connected to consecutive ports as needed.
2. Connect a CameraLink cable from the CameraLink #1 connector on the back panel of the Camera Controller to the #1 Connector of your CameraLink frame grabber.
3. If necessary, connect a second CameraLink cable from the CameraLink #2 connector on the back panel of the Camera Controller to the #2 Connector on the frame grabber.
4. Connect the Camera Controller to the power supply.
5. If using an EC 11000 or EC 16000, connect the Camera to the auxillary power supply. The Thermal Electronic Cooler (TEC) and Fan will not work unless the Camera is plugged into a separate power source.

NOTES:

1. Do not combine EC 11000 Camera Heads with other Camera Head models on the same Camera Controller.
2. Depending on your camera configuration and frame grabber, you may need a second CameraLink cable.

6.3.2 Serial Communications link and the CameraLink Interface

If you want to use the serial communications link that is embedded in the CameraLink interface, the frame grabber vendor will supply a dynamic link library (DLL) that provides access to the serial port on the frame grabber. The format and API for this library are specified by the CameraLink standard. Your frame grabber software installation should place a file in your Windows system directory named clser***. Dll, where *** is a unique three-letter identifier for the frame grabber vendor.

The CameraLink DLL should already be present on your computer. The CameraLink serial driver can be specified during the *MEGAPLUS* Central software camera installation process from a pull-down menu in the Camera Installation dialog.

Some frame grabber vendors provide a method for using the frame grabber's on-board serial communications hardware as a standard COM port available to the operating system (for example COM 9). A virtual serial port is created that uses the frame grabber's proprietary serial interface. If your frame grabber vendor supports this feature, the new virtual serial port may be created by using their software. You may configure *MEGAPLUS* Central to communicate with the camera through the newly created virtual COM port.

Related information:
See "CameraLink Frame Grabber Installation" ""on page 32

6.4 Camera Controller Front and Back Panel Connectors

6.4.1 Front Panel Connectors

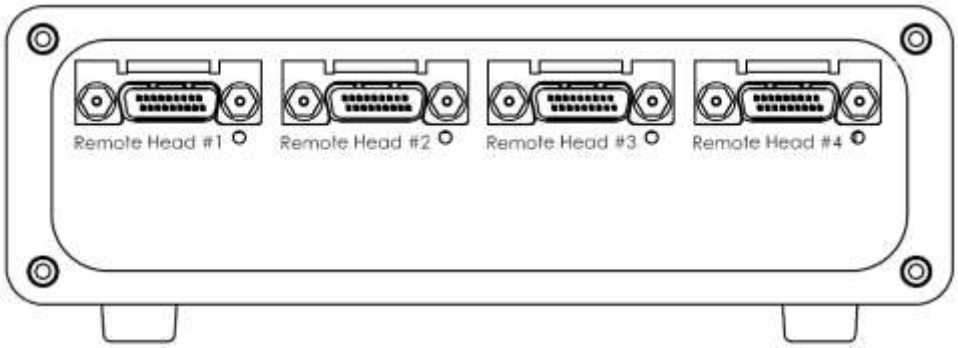


Figure 6.1 Multi-Head Controller Front Panel

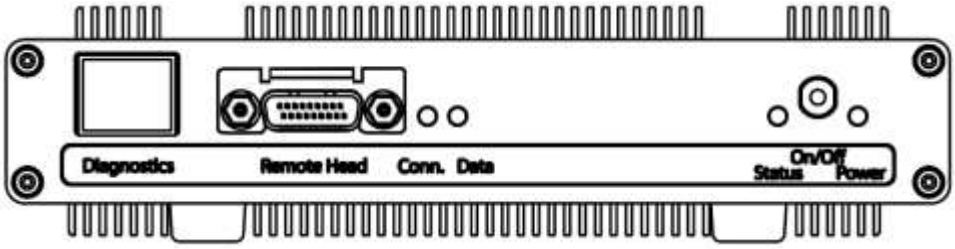


Figure 6.2 Single-Head Controller Front Panel

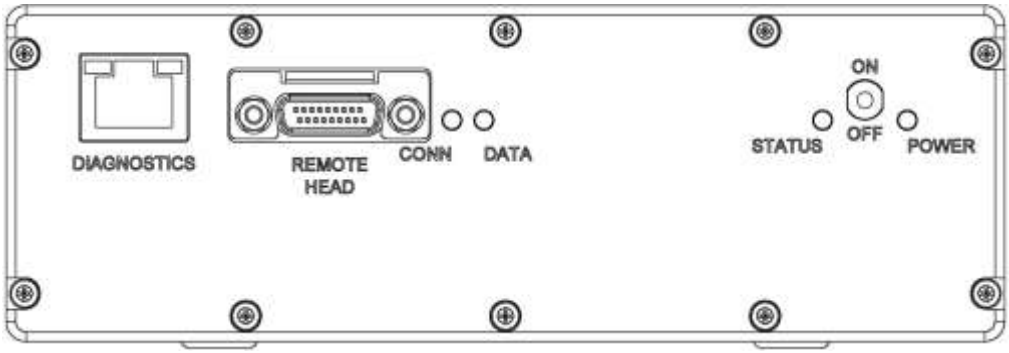


Figure 6.3 GigE Vision Single-Head Controller Front Panel

Remote Head(s): The Camera Heads connect to the Camera Controller via Princeton Instruments factory supplied cables. The Camera Controller may have either one or four Remote Head connectors. A lit green LED indicates a Camera Head is connected to the Camera Controller.

Conn and Data LEDs: These LEDs indicate the connection status and data transfer between the controller/camera with the host computer.

On/Off Switch: Toggle switch.

Status and Power LEDs: These LEDs indicate when the Controller is powered on and its initialization status.

Diagnostics: This100Base-T Ethernet port is designed for field upgrades of camera firmware.

CAUTION! Make sure the MEGAPLUS Camera Controller power switch is turned to the OFF position before connecting the cables!

6.4.2 Back Panel Connector

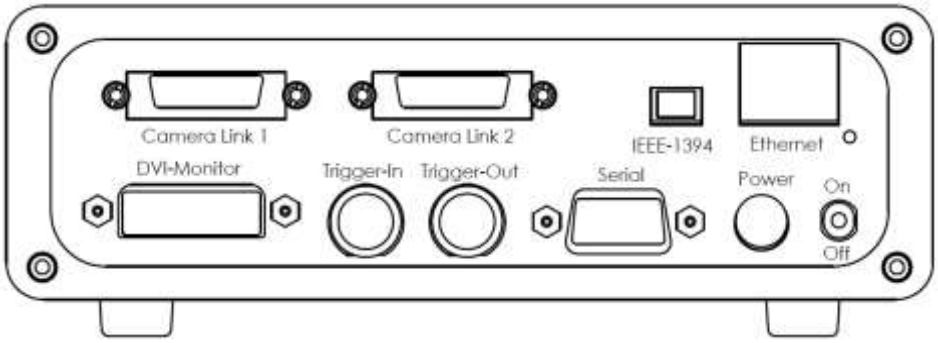


Figure 6.4 Multi-Head Controller Rear Panel

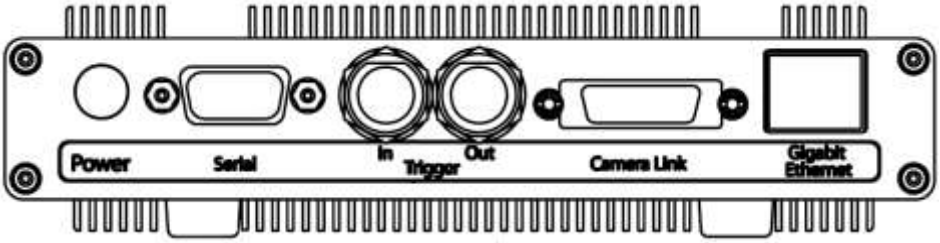


Figure 6.5 Single-Head Controller Rear Panel

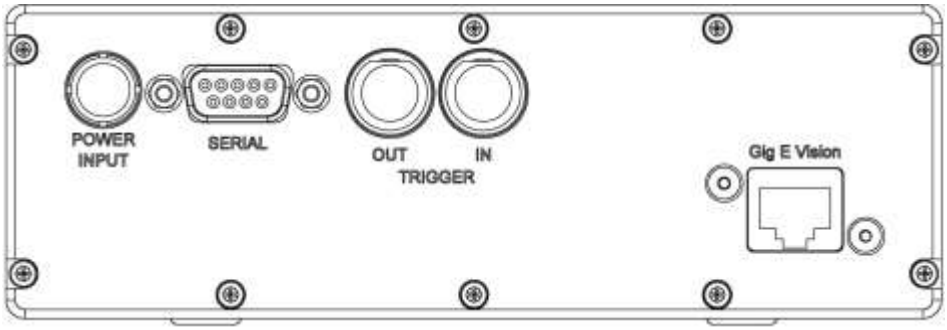


Figure 6.6 GigE Vision Single-Head Controller Rear Panel

Camera Link/Camera Link 1 (Base): Used to connect the controller with a CameraLink frame grabber. The CameraLink serialized frame grabber interface is compliant with the industry standard CameraLink Specification. This specification is available on the Automated Imaging Association website (**www.machinevisiononline.org**).

Camera Link 2 (Med): Located on the rear of the multi-head controller, this connector enables limited use of the Camera Link interface.

IEEE 1394a -4-pin: (multi-head controller only) FireWire connector enabling IIDC compliant image transfer and camera control. The FireWire interface is compliant with the IEEE Standard 1394a-2000 specification.

Ethernet: (multi-head controller) A 100Base-T Ethernet port designed for field upgrades of camera firmware.

Gigabit Ethernet: (Non-GigE Vision single-head controller) Located on the rear of the single-head controller, this functionality has not been implemented.

GigE Vision: (single-head GigE Vision controller only) Located on the rear of the GigE Vision single-head controller, this allows you to connect a Cat5e or Cat6 Ethernet cable to a dedicated Ethernet port in the host computer.

DVI: (multi-head controller) This port allows you to output video via the DVI port of the *MEGAPLUS* Camera Controller. This interface is compliant with the Digital Display Working Group, DDWG, Digital Visual Interface specification, Revision 1.0). Adapters are available to attach the DVI connector to the 15-pin Sub-D VGA port found on many computer monitors. Software controls for DVI output are available in MEGAPLUS Central version 1.30 and greater.

WARNING! DO NOT use the DVI output option for extended periods on Camera Controllers that are not equipped with an integrated cooling fan. Using the DVI output option generates significant heat inside the Camera Controller.

Trigger-In: A BNC connector for external trigger of the controller.

Trigger-Out: A BNC connector which provides a trigger out for synchronization with connected data acquisition equipment.

Serial: An RS232 serial port which can be used for camera control.

Power: A Lemo EPG.0B.302.HLN receptacle.

On/Off Switch: Located on the rear of the multi-head controller. This toggle switch allows you to turn the controller on or off.

Related Information:
See "Connectors, Pin Outs & Cables" on page 159.

This page intentionally left blank.

7. Software Installation

7.1 Firewire

7.1.1 Overview

- IEEE 1394a applies to multi-head controller only.
- Make sure your IEEE 1394a interface hardware is properly installed on your host computer. Follow the hardware supplier's instructions. Also make sure you have connected the Camera Head and PC to the Camera Controller.
- Install the NI-Vision Acquisition Software. This driver is necessary for the *MEGAPLUS* Central Camera Control Software and the MEGAPLUSLib DLL to access the 1394 data communications bus.
- Open the Measurement and Automation Explorer (MAX) program and turn on the camera to verify that the camera is properly identified by the IEEE 1394a hardware and assigned to the proper system driver.
- Install the *MEGAPLUS* Central Camera Control software from the *MEGAPLUS* software CD that was included with your camera.
- Optional: If you plan to write programs for camera control using the *MEGAPLUS* Software Developer's Kit (SDK), install the SDK software.
- Verify camera operation with *MEGAPLUS* Central.

7.1.2 Install the NI Vision Acquisition Software

1. Insert the NI Vision Acquisition Software CD into your CD drive. The installation program should automatically start. If your system does not start the installation, open Windows Explorer and double-click on the setup.exe application on the CD.
2. Select the Install NI Vision Acquisition Software option from the installation screen.

Figure 7.1 NI-Vision Acquisition Software Install Screen

NOTE: During the installation, all 1394 cameras should be unplugged from the system and you should exit any other programs that are running.

3. Follow the installation steps until you come to the Features screen shown in Figure 7.2

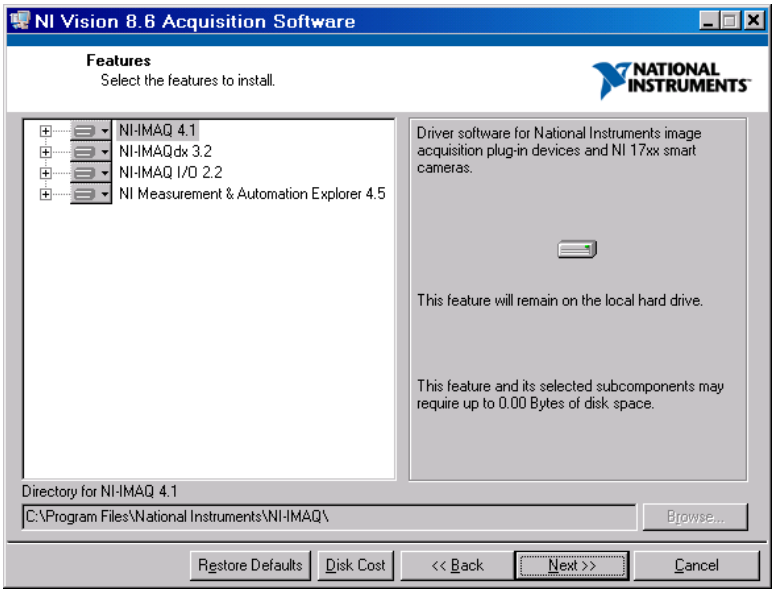

NI Vision 8.6 Acquisition Software
Features
Select the features to install.
NATIONAL INSTRUMENTS
NI-IMAQ 4.1
NI-IMAQdx 3.2
NI-IMAQ I/O 2.2
NI Measurement & Automation Explorer 4.5
Driver software for National Instruments image acquisition plug-in devices and NI 17xx smart cameras.
This feature will remain on the local hard drive.
This feature and its selected subcomponents may require up to 0.00 Bytes of disk space.
Directory for NI-IMAQ 4.1
C:\Program Files\National Instruments\NI-IMAQ\
Browse...
Restore Defaults
Disk Cost
<< Back
Next >>
Cancel

Figure 7.2 NI Vision Features Dialog

4. In the expanding menu there are a number of selections listing support for specific programming or application environments. The entries marked with a disk drive symbol will be installed. Any entry marked with an "X" will not be installed. Adjust the entries on the dialog as follows:
    - If you do not plan to use any programming tools, you can deselect each of the entries that are labeled Support for*** by clicking on the disk drive to change the selection to Do Not Install.
    - If you will be using any of the specified programming tools, select the appropriate entries and change the setting to Install this Feature to a local drive.
    - If you are not certain, the most common selections are Support for Microsoft Visual C and Support for Microsoft Visual Basic. Make sure the proper driver files, support files, and documentation entries for your configuration are selected for installation.
5. Click on the Next button to proceed to the next step of the installation. Follow the remaining on-screen instructions until the installation is complete.

7.1.3 Verify Camera Identification and Driver Assignment

1. Find the Measurement and Automation Explorer icon on the desktop and double click to start the program; or from the Windows taskbar, select: Start > All Programs > National Instruments > Measurement and Automation.
2. Click on the "+" to expand the Devices and Interfaces section.
3. Turn on the camera by switching the Camera Controller power on the rear panel to the ON position.

NOTE: Wait approximately 45 seconds for the camera to complete its "boot" process. The green LED on the rear panel of the Camera Controller will blink at a steady rate (about 1 blink per second) when the camera is ready to operate.

4. Click on the "+" to expand the Legacy NI-IMAQ IEEE 1394 Devices section. An entry for the camera should now appear in the list.

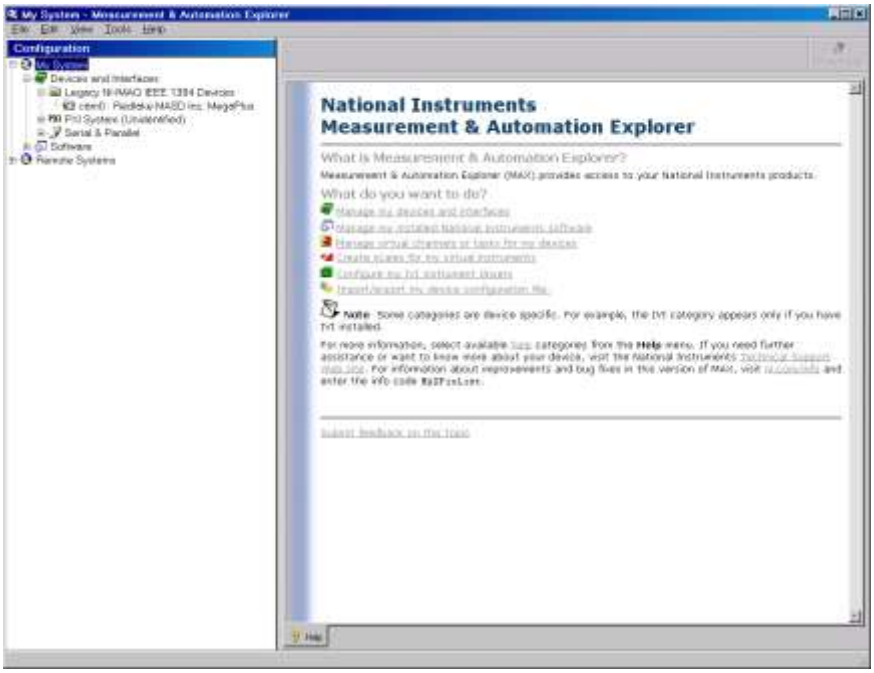


Figure 7.3 Measurement and Automation Explorer Window

5. If a yellow exclamation mark displays next to the camera or it is listed as Generic 1394 Desktop Camera, position the mouse over the camera entry and right click. From the pop-up menu that displays, select Driver and then select the Legacy NI-IMAQ IEEE 1394a IIDC Digital Camera.

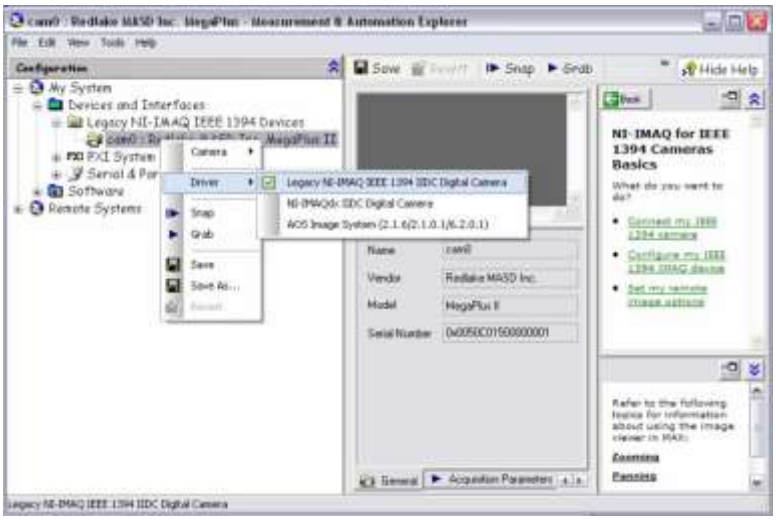


Figure 7.4 Driver Menu

6. A warning about changing the driver will appear and you are asked to confirm that you want to update the driver. Click on the Yes button.
7. When the camera is properly assigned to the IMAQ for 1394 driver, the Princeton Instruments *MEGAPLUS* camera entry will display.
8. Click on the Princeton Instruments *MEGAPLUS* entry and the Measurement and Automation Explorer window should display as shown in Figure 7.5.

For more troubleshooting information on NI-IMAQ, see "Troubleshooting" on page 197.

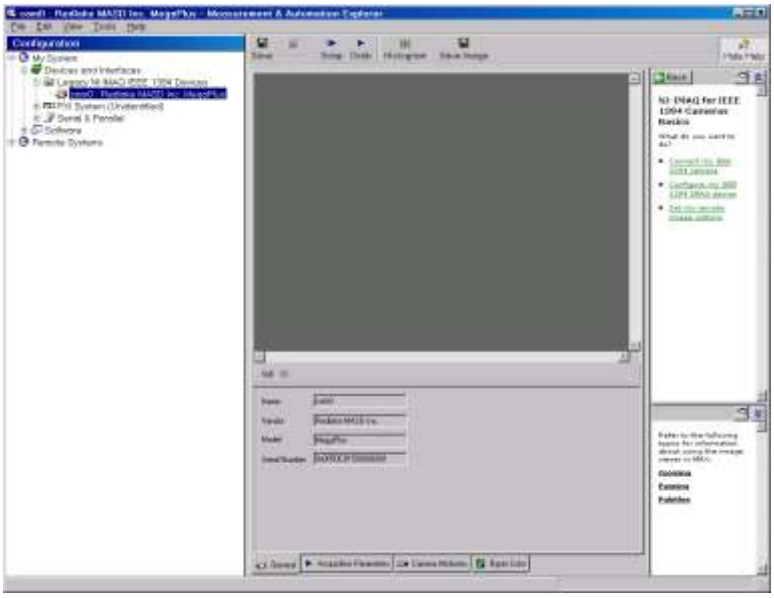


Figure 7.5 MAX Window when *MEGAPLUS* Camera is Open

9. Close the Measurement and Automation Explorer. If a message box appears asking if the current camera configuration should be saved, click on the Yes button.

NOTE: To find out more about using NI Explorer for image acquisition and testing, please refer to the NI Measurement and Automation Explorer on-line help and the documentation included on the NI CD.

7.1.4 Install the *MEGAPLUS* Central Software

1. Insert the *MEGAPLUS* Software CD into your CD drive. The menu program should automatically start as shown in Figure 7.6. If your system does not start the menu, open Windows Explorer, go to the Install folder on the CD and double-click on the "MenuBox.exe" application on the CD.

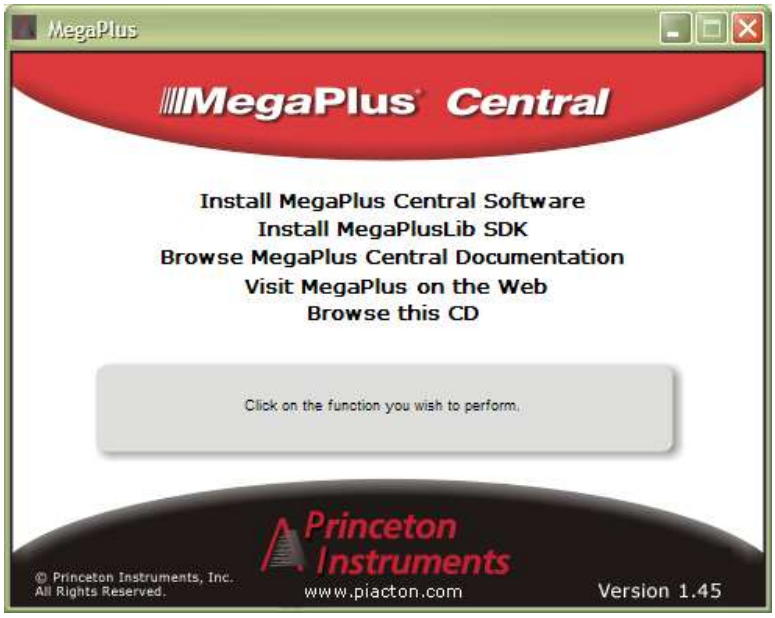


Figure 7.6 *MEGAPLUS* Software CD Menu

2. Select the option to Install *MEGAPLUS* Central Software and follow the on-screen instructions.

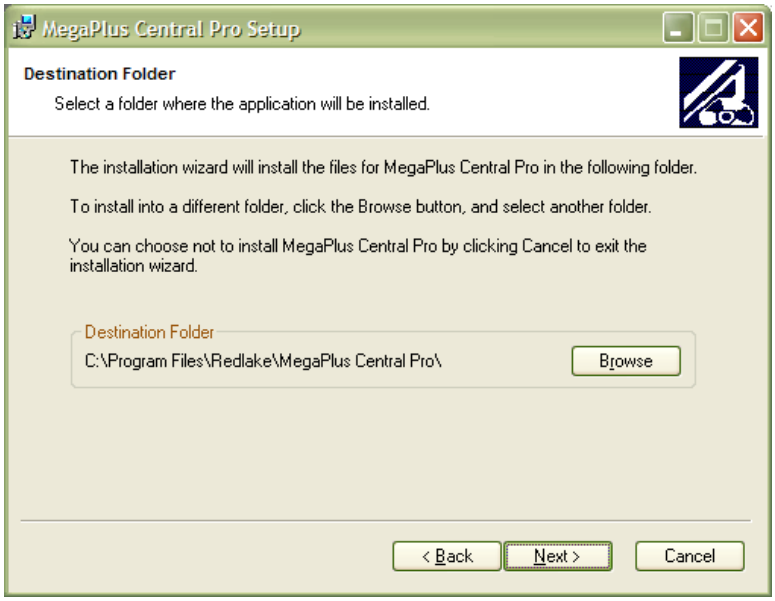

MegaPlus Central Pro Setup
Destination Folder
Select a folder where the application will be installed.
The installation wizard will install the files for MegaPlus Central Pro in the following folder.
To install into a different folder, click the Browse button, and select another folder.
You can choose not to install MegaPlus Central Pro by clicking Cancel to exit the installation wizard.
Destination Folder
C:\Program Files\Redlake\MegaPlus Central Pro\
Browse
< Back
Next >
Cancel

Figure 7.7 *MEGAPLUS* Central Setup Program

When the application files have been installed, the MEGAPLUS Camera Installer application may start automatically as shown in Figure 7.8. In order for the Princeton Instruments *MEGAPLUS* software to access your camera, it must be described in the system camera list. The Camera Installer application adds an entry for your *MEGAPLUS* camera into the list.

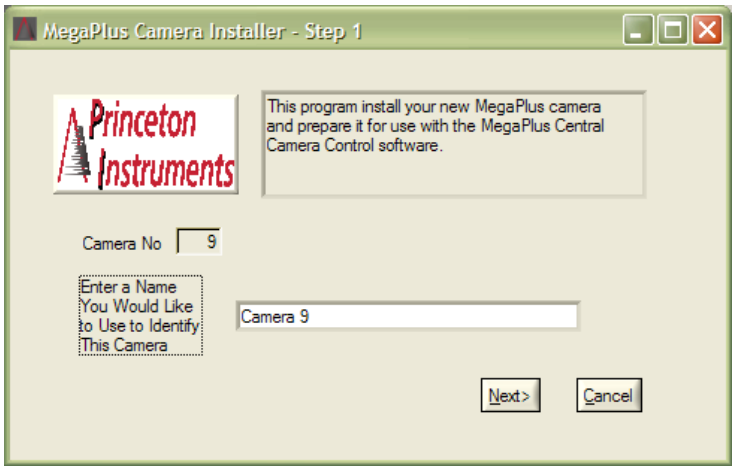

MegaPlus Camera Installer - Step 1
Princeton Instruments
This program install your new MegaPlus camera and prepare it for use with the MegaPlus Central Camera Control software.
Camera No
9
Enter a Name You Would Like to Use to Identify This Camera
Camera 9
Next>
Cancel

Figure 7.8 *MEGAPLUS* Camera Installer - First Screen

3. You can enter a descriptive name that will help to identify the camera if you have multiple cameras in use. A default camera number identifies each camera in the *MEGAPLUS* Camera List.

4. For each camera, you must specify the data interface that you intend to use with the camera. The *MEGAPLUS* Central Software will support one only one given data interface at a time. You can easily switch to a different interface through the Camera List option in *MEGAPLUS* Central.

Figure 7.9 Specifying the Camera Data Interface

For use with a frame grabber you may select these options:

COM Port Serial - Select this interface if you will be using the DB9 connector on the rear panel of the Camera Controller and a separate serial cable to control the camera. Specify the COM port you are connected to on the host. In the Windows Device Manager, enter the following port settings for the specified COM port:

- 9600 baud
- 8 data bits
- 1 stop bit
- no parity
- no flow control

CameraLink Serial - Select this interface if you will be using a CameraLink frame grabber and want to use the CameraLink serial port. This option requires that your vendor supply the clser***. Dll serial port API as specified by the October 2000T CameraLink standard. After you have selected the CameraLink interface option, you must select your vendor's DLL from the pull-down list.

5. Specify the camera control and image acquisition interface to use with the camera. To use 1394, select the FireWire 1394 entry. The *MEGAPLUS* Central software will support only one given data interface at a time. Switch to a different interface through the Camera List option in *MEGAPLUS* Central.
6. Click on the Next button and then review the selections before clicking on the Finish button.

NOTE: With the exception of Gigabit Ethernet (GigE Vision), you may have multiple cameras in your Camera List file. If you have previously run the Camera Installer, or there are other cameras on the system there may be more than one entry in your camera file. This is not harmful. In fact, you may decide to define more than one description of your camera for different interfaces. You can quickly switch between these configurations by selecting File > System Camera List in MEGAPLUS Central software.

7.1.5 Start Your Camera and Software

At this point, the installation process is complete. You may begin using your camera immediately.

1. Close the CD Installation Menu by clicking on the Close box in the upper right corner.
2. Verify that the green Status LED is blinking at a steady rate of about 1 blink per second. This indicates that the camera has finished initializing and that it is ready to acquire images. (When the camera is turned on, it takes about 45 seconds to complete the boot process.)

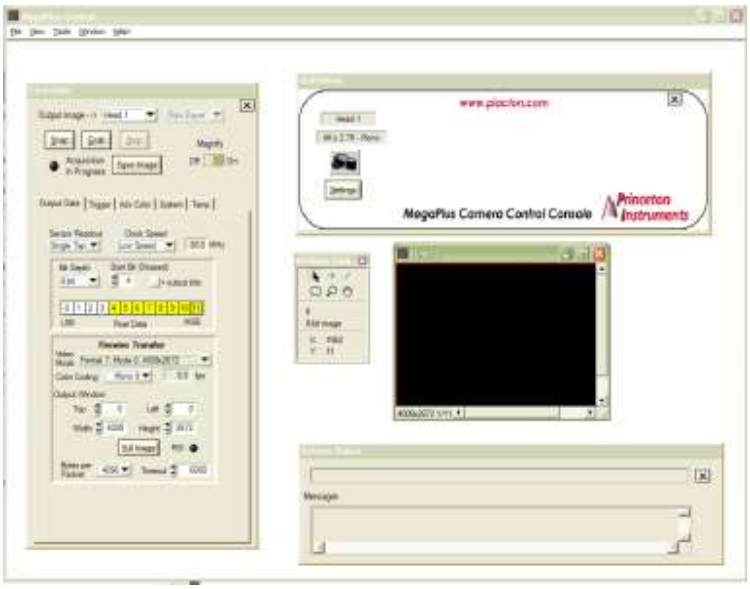


Figure 7.10 *MEGAPLUS* Central Main Screen

3. Start running *MEGAPLUS* Central. The program queries the camera to identify its current configuration (how many heads are attached and what type they are) and then configures itself appropriately.

The Controller Control panel provides access to system level features.

To control the Camera Head, click on the Setting button in the Camera Control panel.

7.2 GigE Vision

7.2.1 Overview:

- Make sure your GigE Ethernet interface hardware is properly installed in your host computer. Follow the computer manufacturer's and hardware supplier's instructions. Also make sure you have connected the Camera and PC to the Camera Controller.
- Install GigE Vision-compliant vision acquisition software such as NI-IMAQdx Vision Software. This is necessary for the *MEGAPLUS* Central Camera Control Software and the MEGAPLUSLib DLL to access the GigE Vision bus. If another GigE Vision Compliant application will be used to control the MegaPlus camera, change its settings, and provide SDK support, then the NI-IMAQdx Vision Software may not be required. Contact your Princeton Instrument representative if further clarification is desired.
- Turn the controller on.

- After initialization finishes, open the Measurement and Automation Explorer (MAX) program, verify that the camera is properly identified by the hardware and assigned to the proper system driver, and acquire data for the first time.
- Install the *MEGAPLUS* Central Camera Control software from the *MEGAPLUS* software CD that was included with your camera.
- Verify camera operation with *MEGAPLUS* Central.

7.2.2 Install the NI-IMAQdx™ Vision Acquisition Software (GigE Vision)

1. Turn off the camera.
2. Insert the NI Vision Acquisition Software CD into your CD drive. The installation program should automatically start. If your system does not start the installation, open Windows Explorer and double-click on the setup.exe application on the CD.
3. Select **Install NI Vision 8.6 Acquisition Software** from the installation screen.

Figure 7.11 NI-Vision Acquisition Software Install Screen

NOTE: During the installation, all cameras should be unplugged from the system and you should exit any other programs that are running.

4. Follow the installation steps until you come to the Features screen shown in Figure 7.2.

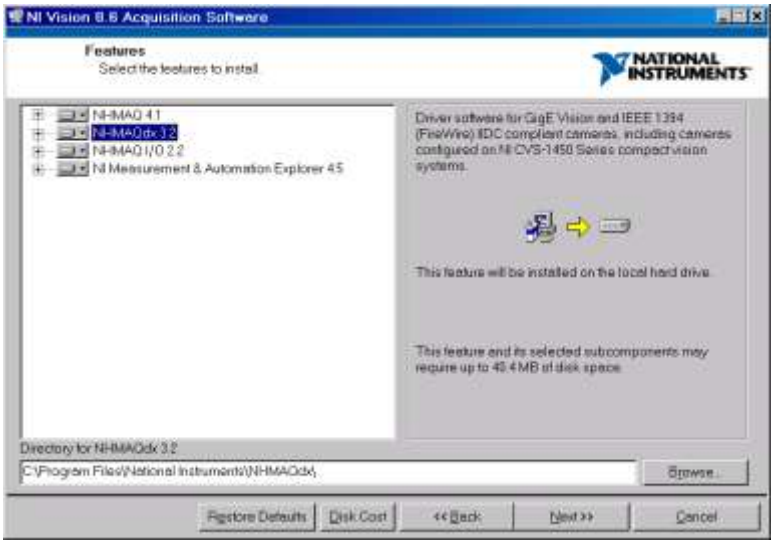
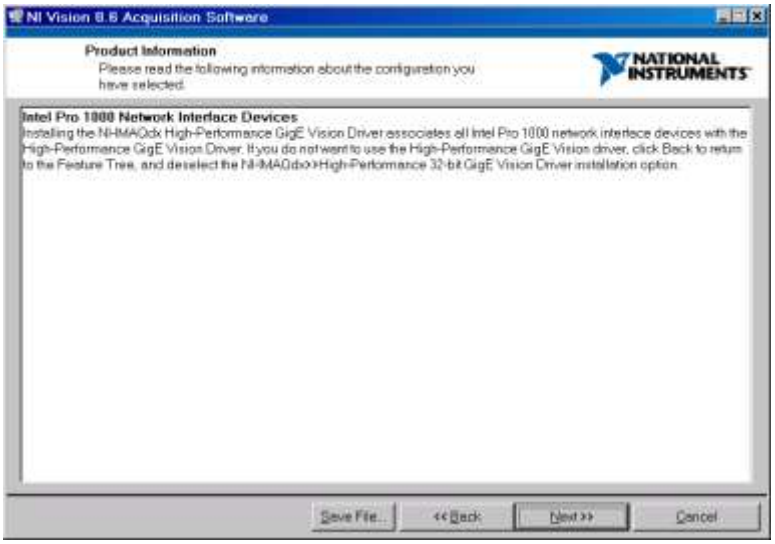

NI Vision 8.6 Acquisition Software
Features
Select the features to install.
NATIONAL INSTRUMENTS
NI-IMAQ 4.1
NI-IMAQdx 3.2
NI-IMAQ I/O 2.2
NI Measurement & Automation Explorer 4.5
Driver software for GigE Vision and IEEE 1394 (FireWire) IIDC compliant cameras, including cameras configured on NI CVS-1450 Series compact vision systems.
This feature will be installed on the local hard drive.
This feature and its selected subcomponents may require up to 40.4 MB of disk space.
Directory for NI-IMAQdx 3.2
C:\Program Files\National Instruments\NI-IMAQdx\
Browse...
Restore Defaults
Disk Cost
<< Back
Next >>
Cancel

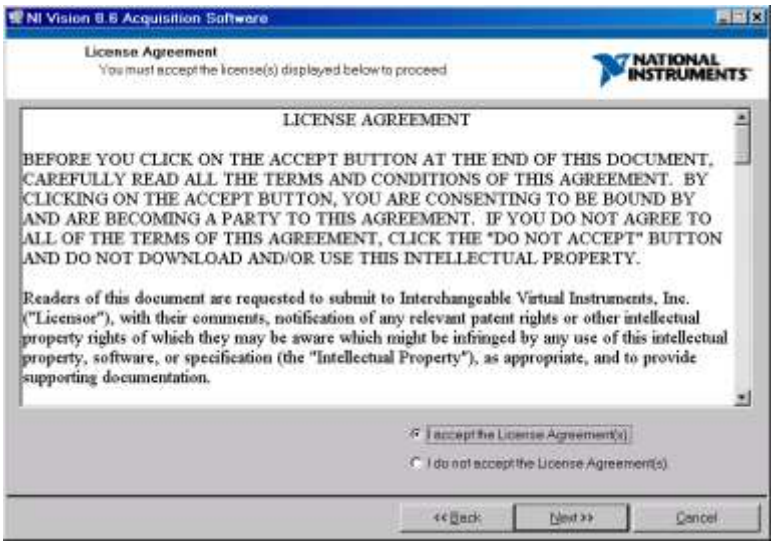
Figure 7.12 NI Vision Features Dialog

5. Select **NI-IMAQdx.32** and click on **Next**.


NI Vision 8.6 Acquisition Software
Product Information
Please read the following information about the configuration you have selected.
NATIONAL INSTRUMENTS
Intel Pro 1000 Network Interface Devices
Installing the NI-IMAQdx High-Performance GigE Vision Driver associates all Intel Pro 1000 network interface devices with the High-Performance GigE Vision Driver. If you do not want to use the High-Performance GigE Vision driver, click Back to return to the Feature Tree, and deselect the NI-IMAQdx>>High-Performance 32-bit GigE Vision Driver installation option.
Save File...
<< Back
Next >>
Cancel

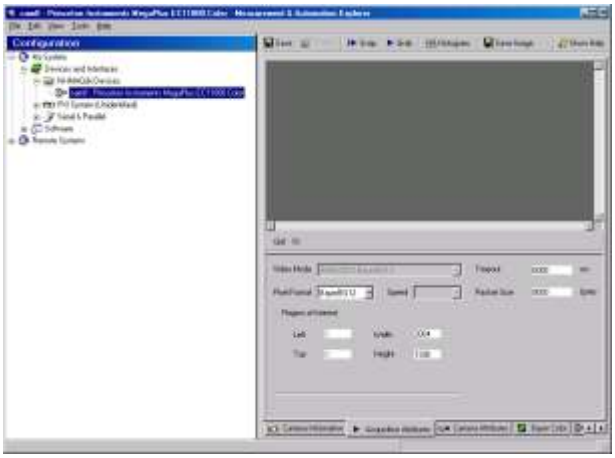
6. Click on **Next** and accept the license agreements.

NI Vision 8.6 Acquisition Software
License Agreement
You must accept the license(s) displayed below to proceed.
NATIONAL INSTRUMENTS
NATIONAL INSTRUMENTS SOFTWARE LICENSE AGREEMENT
INSTALLATION NOTICE: THIS IS A CONTRACT. BEFORE YOU DOWNLOAD THE SOFTWARE AND/OR COMPLETE THE INSTALLATION PROCESS, CAREFULLY READ THIS AGREEMENT. BY DOWNLOADING THE SOFTWARE AND/OR CLICKING THE APPLICABLE BUTTON TO COMPLETE THE INSTALLATION PROCESS, YOU CONSENT TO THE TERMS OF THIS AGREEMENT AND YOU AGREE TO BE BOUND BY THIS AGREEMENT. IF YOU DO NOT WISH TO BECOME A PARTY TO THIS AGREEMENT AND BE BOUND BY ALL OF ITS TERMS AND CONDITIONS, CLICK THE APPROPRIATE BUTTON TO CANCEL THE INSTALLATION PROCESS, DO NOT INSTALL OR USE THE SOFTWARE, AND RETURN THE SOFTWARE WITHIN THIRTY (30) DAYS OF RECEIPT OF THE SOFTWARE (INCLUDING ALL ACCOMPANYING WRITTEN MATERIALS, ALONG WITH THEIR CONTAINERS) TO THE PLACE YOU OBTAINED THEM. ALL RETURNS SHALL BE SUBJECT TO NI'S THEN CURRENT RETURN POLICY.
1. Definitions. As used in this Agreement, the following terms have the following meanings:
A. "Academic Institution" Means a degree-granting educational institution.
I accept the License Agreement(s).
I do not accept the License Agreement(s).
<< Back
Next >>
Cancel


NI Vision 8.6 Acquisition Software
License Agreement
You must accept the license(s) displayed below to proceed
NATIONAL INSTRUMENTS
LICENSE AGREEMENT
BEFORE YOU CLICK ON THE ACCEPT BUTTON AT THE END OF THIS DOCUMENT, CAREFULLY READ ALL THE TERMS AND CONDITIONS OF THIS AGREEMENT. BY CLICKING ON THE ACCEPT BUTTON, YOU ARE CONSENTING TO BE BOUND BY AND ARE BECOMING A PARTY TO THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL OF THE TERMS OF THIS AGREEMENT, CLICK THE "DO NOT ACCEPT" BUTTON AND DO NOT DOWNLOAD AND/OR USE THIS INTELLECTUAL PROPERTY.
Readers of this document are requested to submit to Interchangeable Virtual Instruments, Inc. ("Licensor"), with their comments, notification of any relevant patent rights or other intellectual property rights of which they may be aware which might be infringed by any use of this intellectual property, software, or specification (the "Intellectual Property"), as appropriate, and to provide supporting documentation.
I accept the License Agreement(s)
I do not accept the License Agreement(s)
<< Back
Next >>
Cancel

7. At the end of the installation, you will be prompted to restart your computer. You must do this before using the software.
8. After you have rebooted your computer, turn on the camera and wait until the Status light is blinking at about 1 blink per second.
9. **This step will be required if you did not purchase the controller and the camera as a system.** With the controller connected via the GigE Vision port to the GigE Vision interface installed in the computer, run the camera.exe driver for your camera. You can find this in the Cameras.zip file located in the MEGAPLUS\Cameras directory. Open the zip file and extract the executable for your camera. The executable filenames are based on the camera name. If you have an EC11000 Mono camera, you would extract the file named EC11000M.exe. Double-click on the executable file and wait until the program finishes updating the firmware in the controller and click on **OK**.
10. Reboot the controller and wait until it finishes initializing.
11. Then start the Measurement & Automation software.

7.2.3 Acquire Data for the First Time

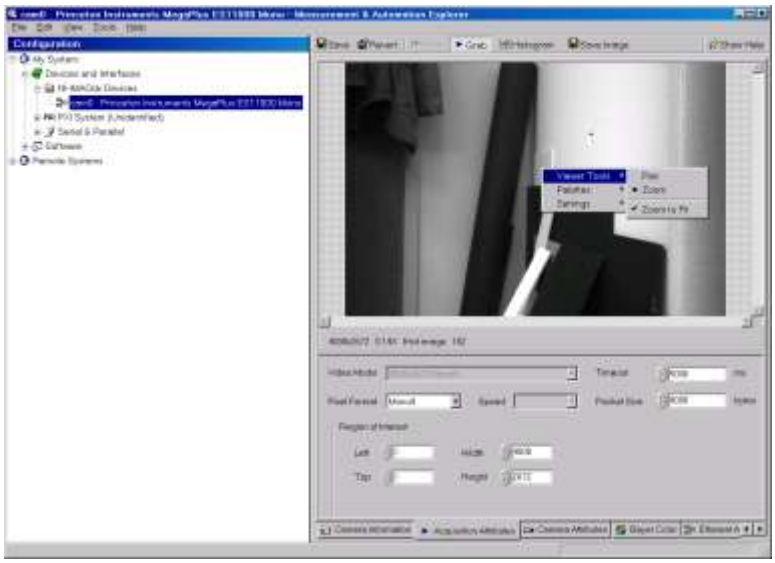


1. Power on the GigE Controller and wait until the "Status" light blinks steadily at about 1 blink per second.
2. Start the NI software named Measurement Automation Explorer (MAX).
3. Click on My System, Devices and Interfaces, and wait until NI-IMAQdx Devices is displayed. When **cam0:PrincetonInstruments MegaPlus…** is displayed, click on it and wait until the data display appears.
4. Select the Acquisition Attributes tab and set the packet size to be the same or smaller than the Jumbo Frames size selected via the Windows Device Manager. For example, if you selected 9014 in Device Manager, you will need to set the packet size to 9012 (the packet size you set in the application software must be

evenly divisible by 4). ***If you don't have an Intel Pro adapter and jumbo packets are not supported, set the packet size to 1500.***

NOTE: The National Instruments GigE Vision Adapter does not allow you to select the frame size and is by default 8000. You can, however, change that to a maximum size of 9012 on the Acquisition Attributes tab.

5. Click on **Snap** or **Grab** and confirm that you can acquire an image.



6. After you've confirmed operation, close MAX. You will be asked if you want to save. Say "Yes."
7. Then close the MAX software.
8. You have now confirmed that the camera can be controlled via the National Instruments® software.

7.2.4 Install the MEGAPLUS Central software

See **Section 7.1.4** for MEGAPLUS Central software installation instructions.

7.2.5 Verify Camera Control with *MEGAPLUS* Central

1. Turn on the camera by switching the Camera Controller power on the rear panel to the ON position.

NOTE: Make sure that your camera is turned on and that you have allowed time for the Camera head to "boot" (about 45 seconds). Check to see that the green light above the power supply at the back of the Camera Head is blinking steadily. If a connection with the camera is not made when the MEGAPLUS Central program starts, the software will automatically retry to establish the connection.

2. Double-click the *MEGAPLUS* Central icon to start the program. During installation, an icon for *MEGAPLUS* Central is placed on the desktop. Or, from the Windows Taskbar select Start > All Programs > *MEGAPLUS* Central.
3. There may be some delay while the program connects to the camera.

Figure 7.13 *MEGAPLUS* Central Gigabit Ethernet Connection

4. After the connection is made, click on the **Snap** button and verify that an image is acquired.

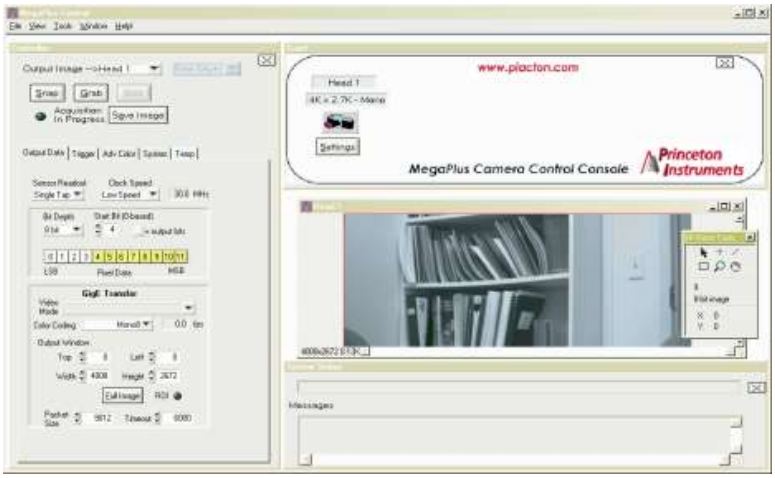


Figure 7.14 *MEGAPLUS* Central Camera Control Window

7.3 CameraLink Frame Grabber Installation

7.3.1 Overview

- Make sure your CameraLink frame grabber hardware and software is properly installed on your host computer. (Follow the frame grabber supplier's instructions).
- Install the *MEGAPLUS* Central Camera Control software from the *MEGAPLUS* Central software CD that was included with your camera.
- Optional: If you plan to write programs for camera control using the *MEGAPLUS* SDK, install the SDK software.
- Connect the Camera Head to the Camera Controller and verify camera control with *MEGAPLUS* Central.
- Verify image acquisition with the frame grabber software using the instructions provided by your frame grabber vendor.

7.3.2 Install Your Frame Grabber Hardware and Software

Before proceeding with the camera software installation, make sure your CameraLink frame grabber hardware and software are installed properly. Follow instructions provided by your frame grabber vendor to verify proper installation.

7.3.3 Verify Camera Control with *MEGAPLUS* Central

1. Turn on the camera by switching the Camera Controller power on the rear panel to the ON position.

   NOTE: Make sure that your camera is turned on and that you have allowed time for the Camera head to "boot" (about 45 seconds). Check to see that the green light above the power supply at the back of the Camera Head is blinking steadily. If a connection with the camera is not made when the MEGAPLUS Central program starts, the software will automatically retry to establish the connection.

2. Double-click the *MEGAPLUS* Central icon to start the program. During installation, an icon for *MEGAPLUS* Central is placed on the desktop. Or, from the Windows Taskbar select Start > All Programs > *MEGAPLUS* Central.

The Controller Control panel provides access to system level features.

To control the camera head, click on the Setting button in the Camera Panel.

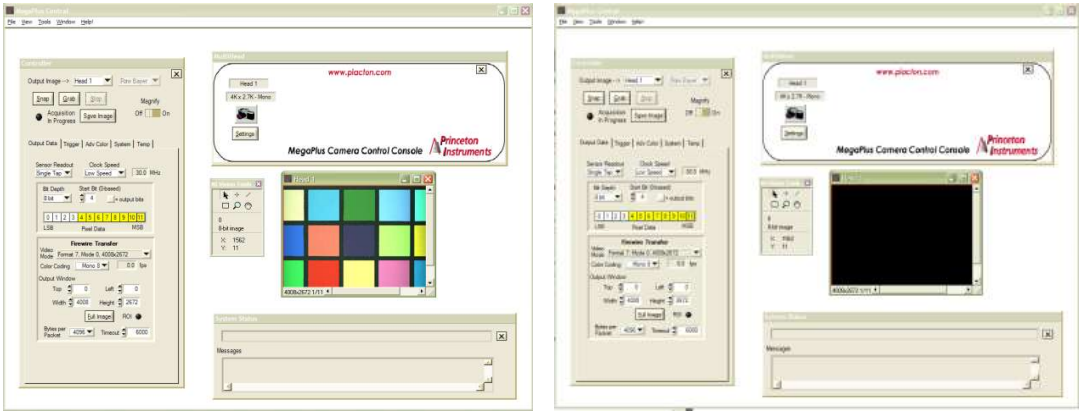


Figure 7.15 *MEGAPLUS* Central Camera Control Window

7.3.4 Verify Image Acquisition with the Frame Grabber Software

Use the software and documentation provided with your frame grabber to begin acquiring images from the camera.

7.4 Install the *MEGAPLUS* SDK (Optional)

The installation is simple and almost identical to installing *MEGAPLUS* Central Software.

NOTE: If you already defined your camera in the MEGAPLUS Camera List, it is not necessary to repeat this step. If you also installed the MEGAPLUS Central program, it is not necessary to repeat the installation of the camera

1. Return to the *MEGAPLUS* installation menu.
2. Select the option to install the *MEGAPLUS* SDK. Follow the on-screen instructions to complete the installation.

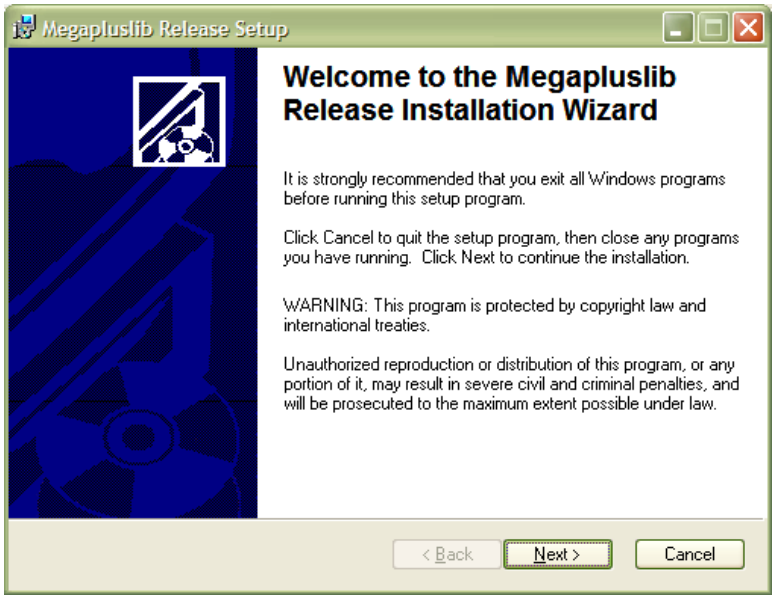

Megapluslib Release Setup
Welcome to the Megapluslib Release Installation Wizard
It is strongly recommended that you exit all Windows programs before running this setup program.
Click Cancel to quit the setup program, then close any programs you have running. Click Next to continue the installation.
WARNING: This program is protected by copyright law and international treaties.
Unauthorized reproduction or distribution of this program, or any portion of it, may result in severe civil and criminal penalties, and will be prosecuted to the maximum extent possible under law.
< Back
Next >
Cancel

Figure 7.16 MEGAPLUS Lib SDK Installation

3. When the installation is complete, the *MEGAPLUS* Camera Installer will start automatically.

NOTE: If you are planning to use a FireWire interface with the MEGAPLUS SDK, the IMAQ IEEE 1394a Camera Driver must be installed before installing the MEGAPLUS SDK.

7.5 How to Change Default Directories for Saving Files

1. From the main toolbar select File > Setup.
2. From the System Settings dialog box, type the new default directories for saving the following files:
   - Images
   - Flat Field Normalization Files
   - Defect Correction Files
   - Color Look up Tables
3. Click on the OK button

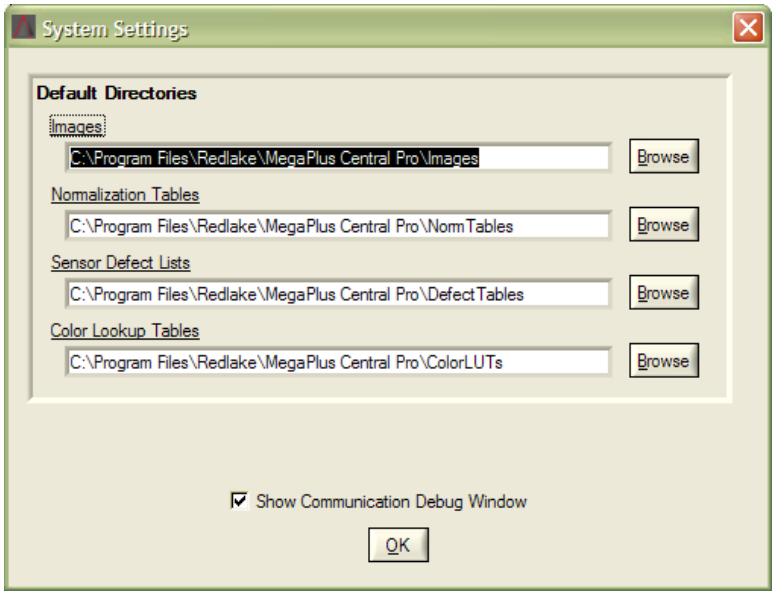


Figure 7.17 Changing Default Directories Dialog Box

This page intentionally left blank.

8. *MEGAPLUS* Central Startup and Overview

1. Double-click the *MEGAPLUS* Central icon to start the program. During installation, an icon for *MEGAPLUS* Central is placed on the desktop. Or, from the Windows Taskbar select Start > All Programs > *MEGAPLUS* Central.
2. When *MEGAPLUS* Central starts, the software checks for the presence of a camera or cameras on the selected communication channel.

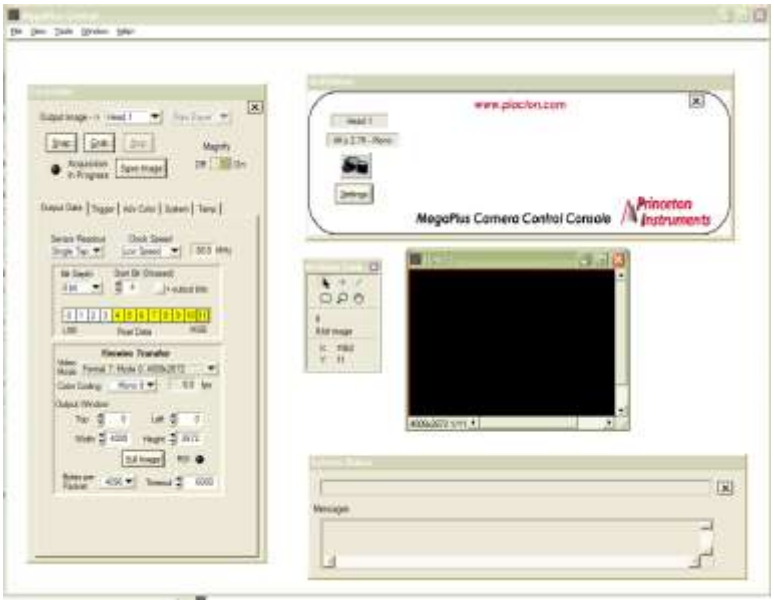


Figure 8.1 *MEGAPLUS* Central Main Window

8.1 System Controls Overview

MEGAPLUS Central has three primary panels for system control:

Controller Control: Use this panel to access the Camera Controller functions.

Camera Control Controller: Displays the name, resolution and type (mono or color) of the available cameras.

Settings Button: Located on the Camera Control Controller, click on this button to access the Camera Head Controls.

System Status Message: Displays error messages.

The display of these panels can be turned on or off using the View tab on the toolbar. The windows can be rescaled using the Windows handles that appear when you click on the panel. The panels within the *MEGAPLUS* Central Window rescale if the window is rescaled.

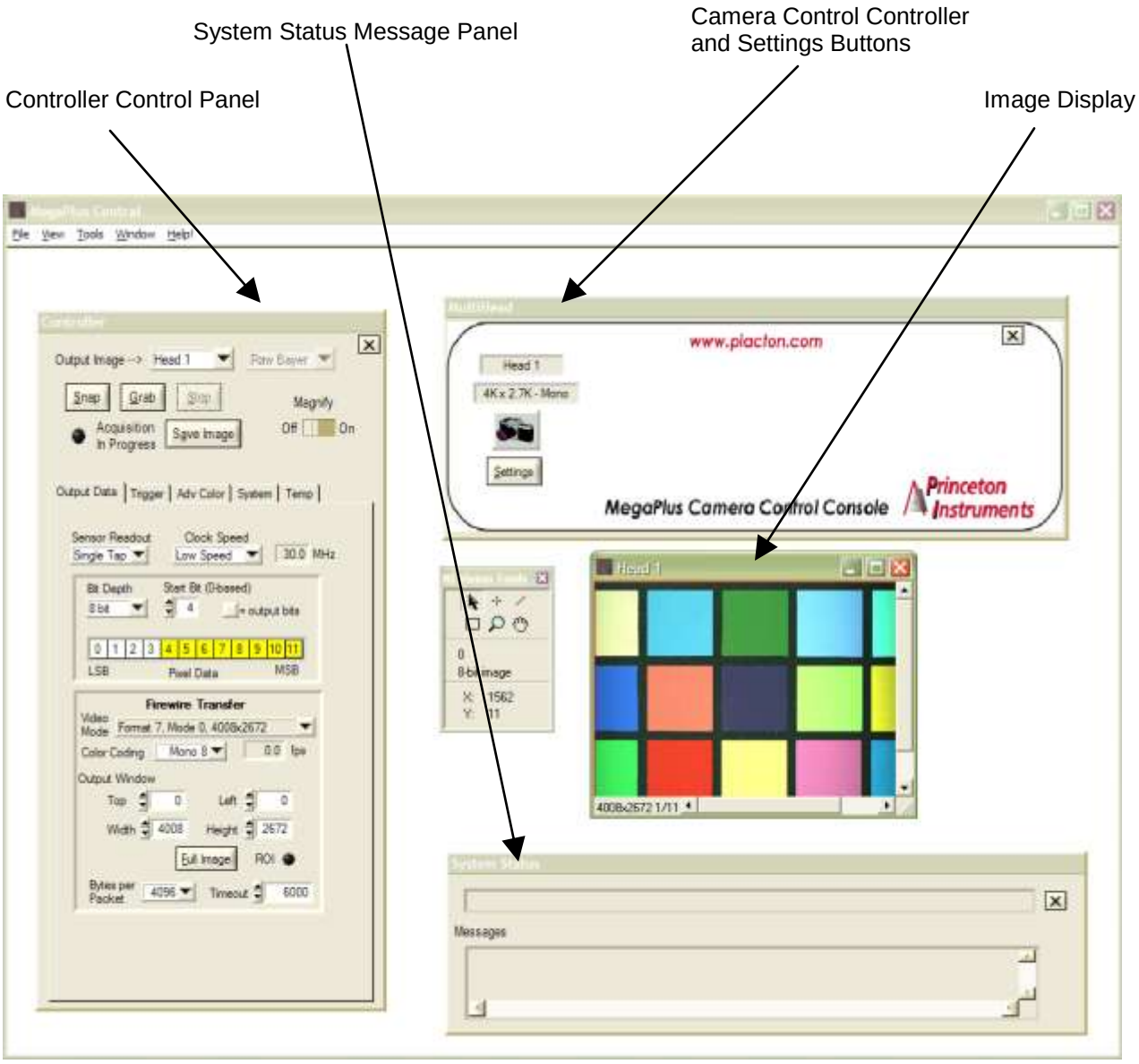


Figure 8.2 *MEGAPLUS* Central Main Window and Serial Debug Window

9. Camera Settings

9.1 How to Set the Gain, Integration Time and Brightness (Offset)

1. Find the desired camera in the Camera Control Controller and click on the Settings button below the camera icon.
2. Click on the Basic tab.
3. Adjust the parameters for Gain, Integration Time and Brightness (Offset) using the following methods:
    - Slide the bar controls to the right or left.
    - Enter a value in the text field.
    - Click on the up or down arrows to change the value in the text field.

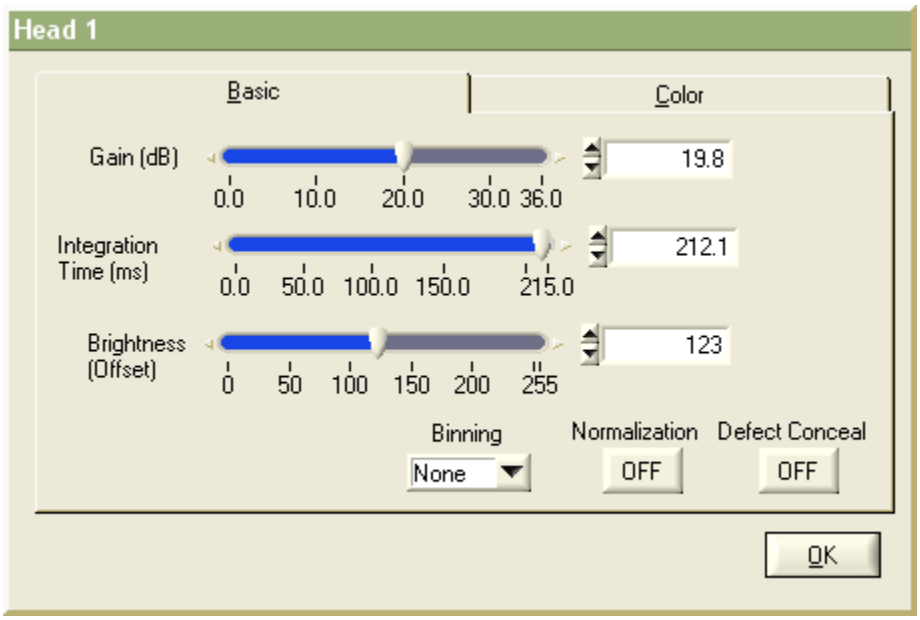


Figure 9.1 Camera Head Control Window- Basic Tab

9.2 Flat Field Normalization

Flat Field Normalization adjusts the value for each pixel to correct for variations in the pixel values and improve the uniformity of the camera's response. To find out more about creating Flat Field Normalization Files contact your Princeton Instruments representative. The *MEGAPLUS* Central software allows the creation, download, and activation of Flat Field Normalization Files.

9.2.1 Why Flat Field Normalization is Needed

In a perfect world, subjecting an ideal sensor to uniform illumination would result in all of the pixels having identical values producing a uniform or flat image. However, in the real world, subjecting a sensor to uniform illumination produces an image with pixel values that vary within a range. These variations in measured signal level arise from a number of sources. One source is random noise, which is due to dark current variations between pixels and gain variations in the signal processing electronics. Other sources of gain variations may occur in systems due to variations in illumination, lens effects, and sensor characteristics.

Some sensors use special technologies such as microlenses to increase sensitivity, but these improvements come with tradeoffs. For instance, microlense technology causes the sensor's quantum efficiency to vary with the angle of incidence of the incoming light. This effect combined with the angular dispersions caused by various lenses may result in a common phenomenon known as shading. Figure 9.2 is an image of an evenly illuminated, flat, white field. The horizontal shading effect is evident in the darker edges and brighter center region of the image. Figure 9.3 is a horizontal profile graph of pixel values versus horizontal positions that quantitatively illustrates the shading effect.

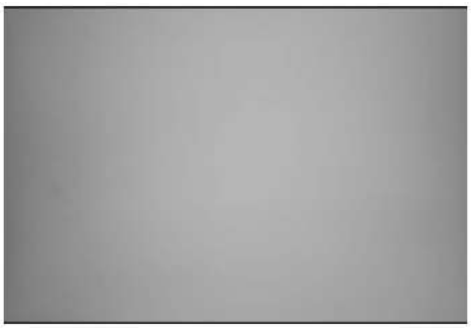


Figure 9.2 Image of an Evenly Illuminated Flat Field with Shading

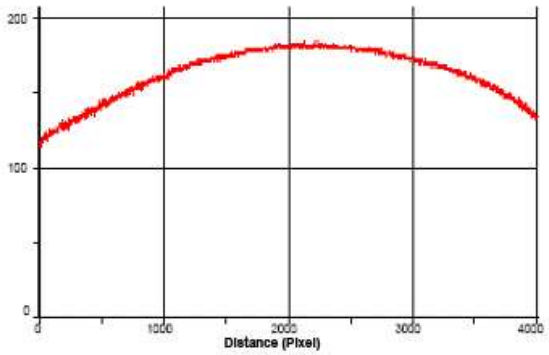


Figure 9.3 Pixel Values at the Horizontal Center of the Image

9.2.2 How Flat Field Normalization Works

The Flat Field Normalization (FFN) process adjusts the value for each pixel to correct for variations in the pixel values and improve the uniformity of the camera's response. MEGAPLUS cameras offer an additive offset and multiplicative gain correction to apply to each pixel in real-time as the image is acquired and processed in the camera.

The normalization memory buffer in the camera contains a set of entries, one for each pixel, which contains a correction factor for both gain and offset. As digital pixel values are acquired from the Camera Head, the unique gain and offset values for the pixel location are extracted from the normalization memory buffer. The offset value is added to the raw pixel value and the resulting value is then multiplied by the gain factor. The resulting value is output from the camera. If the pixel location is marked as a defective pixel, the pixel value is replaced using the defect masking process. MEGAPLUS stores up to four correction tables to meet the needs of more than one configuration.

MEGAPLUS Central "Pro" includes an analysis module for analyzing Camera Head performance as well as building and editing flat field normalization tables. The normalization calculation process builds a data file that specifies a unique gain and offset value that applies to the correction of each pixel in a specific Camera Head.

The normalization table calculation requires dark field and flat field reference images captured from the camera when it is operating in the application environment.

The dark field image is acquired with no light to the sensor. This image is used to create the offset map by calculating the difference between the actual pixel value and the average pixel value at each location. The MEGAPLUS supports offset corrections of up to ±3 DN (digital numbers) per pixel.

The flat field image is used to create the gain map by comparing each pixel to a target value. The target value for each pixel is created by multiplying a gain factor by a given pixel value. The MEGAPLUS supports gain corrections of X1 to X4 (with 12-bit resolution) per pixel.

9.2.3 Flat Field Normalization with CameraLink or COM Port

The MEGAPLUS Central software does not allow image capture from third party frame grabbers or by using only a COM port. However, MEGAPLUS Central Pro version 1.45 or later supports Flat Field Normalization table creation from reference images acquired by frame grabber software packages. *The reference images must be in 16-bit TIF format (12-bit data saved as 16 bits per pixel and must be of "Little Endian").*

Before You Begin

- Install the NI-IMAQ IEEE IMAQ Drivers (required for offline FFN file creation).
- Install MEGAPLUS Pro version 1.45 or higher.
- Install frame grabber software capable of capturing 12-bits per pixel. The software should be capable of saving the image in 16-bit TIF format ("Little Endian" type).
- Make sure a crossover Ethernet cable is connected between the controller and the PC.

Overview of Steps

- Capture a dark field reference image (using third-party frame grabber software).
- Capture a flat field reference image (using third party-frame grabber software).
- Load the dark field and flat field reference images into the Normalization Calibration window.
- Calculate the target pixel value.
- Analyze the reference images to calculate the gain and offset values.
- Download the table to the camera.
- Reset the controller to load the new tables.
- Turn correction on or off for each Camera Head.

Figure 9.4 shows the Normalization Calibration window from which Flat Field Normalization procedures are done.

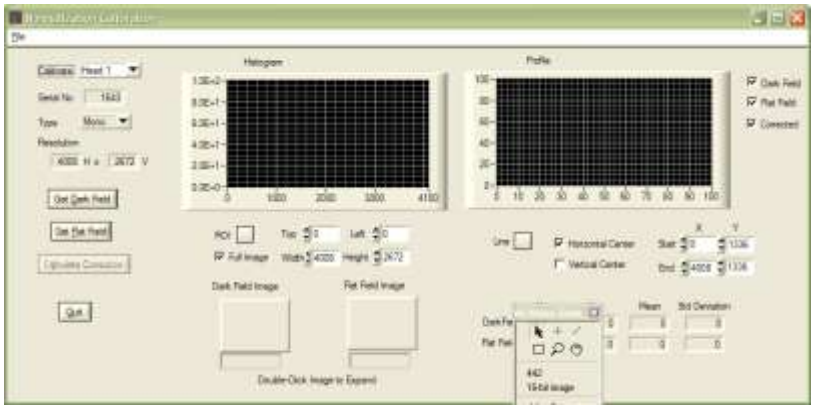


Figure 9.4 Normalization Calibration Window

Capture a Dark Field Reference Image

Software from a third party frame grabber or customer-generated software is required to acquire the appropriate images. Refer to the documentation of your frame grabber for more information.

1. Set the bit depth to 12-bits.
2. Place the lens cap on the camera head. If desired, change the gain and/or integration time.
3. Capture an image.
4. Save the image in 16-bit TIF format. Note: *The TIF format must be of "Little Endian".* Please refer to your capture software for more information.

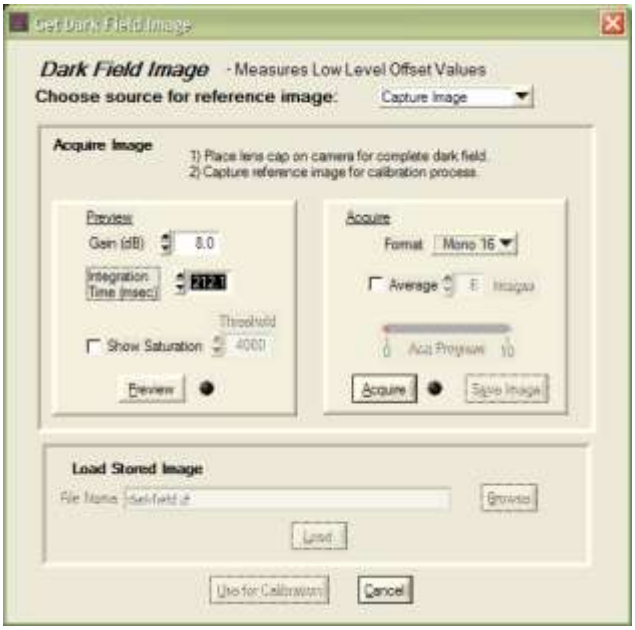


Figure 9.5 Get Dark Field Image Dialog Box

Capture a Flat Field Reference Image

Software from a third party frame grabber or customer-generated software is required to acquire the appropriate images. Refer to the documentation of your frame grabber for more information.

1. Use the same lens, f-stop setting, and external light source that will be used in the application.
2. Uniformly illuminate the target.
3. Adjust the light intensity, lens aperture, and integration time to produce an average pixel value about 80 percent of full scale.
4. Capture an image.
5. Save the image in 16-bit TIF format. Note: *The TIF format must be of "Little Endian".* Please refer to your capture software for more information.

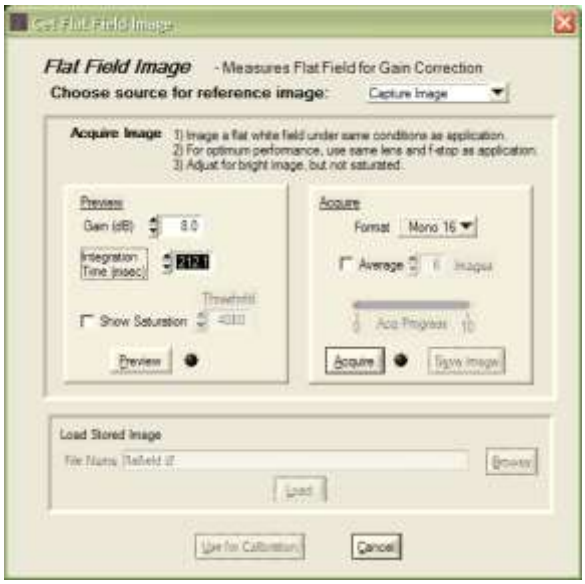


Figure 9.6 Get Flat Field Image Dialog Box

Load Dark Field and Flat Field Images

1. Click on 'Get Dark Field' on the Normalization Calibration window.
2. Select the captured dark field reference image under 'Load Stored Image' and click 'Load' and then 'Use for Calibration'.

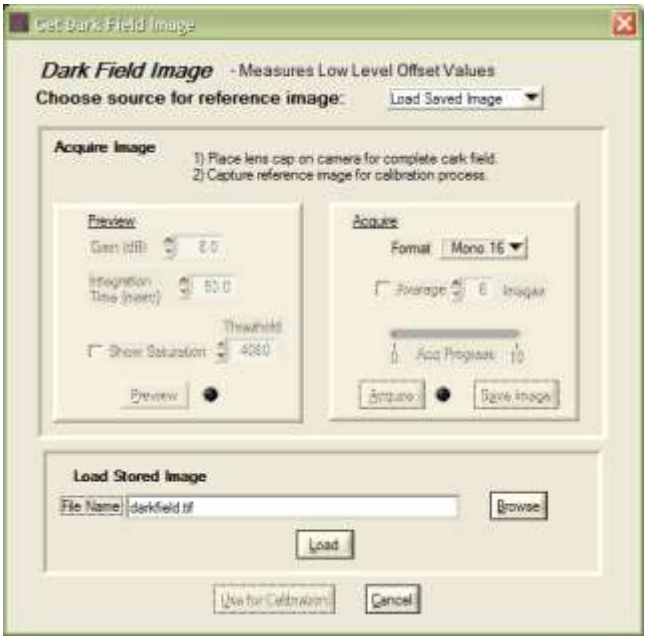


Figure 9.7 Get Dark Field Image Dialog Box

3. Click on 'Get Flat Field' on the Normalization Calibration window.
4. Select the captured flat field reference image under 'Load Stored Image' and click 'Load' and then 'Use for Calibration'.

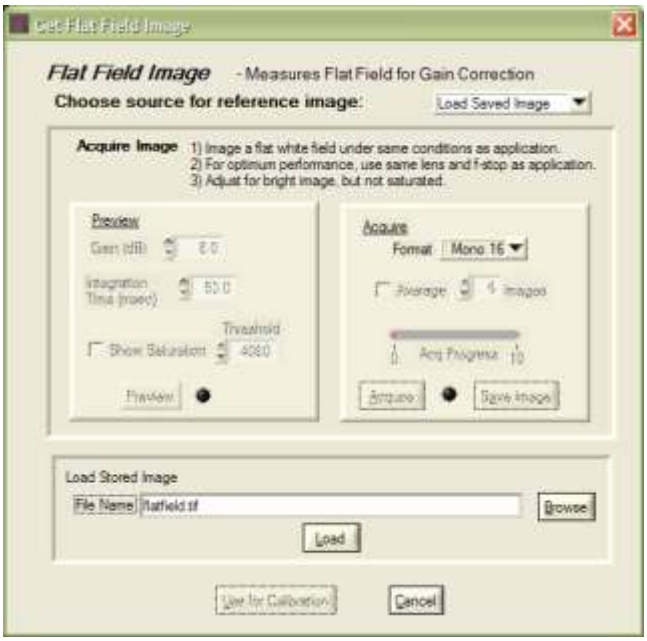


Figure 9.8 Get Flat Field Image Dialog Box

Calculate the Correction

Figure 9.9 Calculate Correction Dialog Box

1. In the Normalization Calibration window, click on the 'Calculate Correction' button.
2. In the Calculate Correction dialog box, type a file name and location in the 'Output File Name' field. The extension must be .ffn.
3. If desired, type in a comment up to 80 characters in the 'Comment for Output File'. At the 'Applies To' drop-down list, select a specific Camera Head serial number to use exclusively with the new table or select the 'Any Serial No' option to use the table with any Camera Head.
4. At the 'Calculate Target With' drop-down list, select the method to derive the target value from the following:
   - Max Pixel Value—the target value will be the maximum pixel value in the ROI of the flat field image.
   - Mean Pixel Value—the target value will be the mean of the pixel values in the ROI of the flat field image.
   - Median Filter and Max—with this method, a median filter is first applied to the ROI in order to smooth the data, and then the maximum pixel value of the resulting data is selected as the target value.
   - Manual Target Value Entry—you may also choose to manually type in the target pixel value.
5. If desired, adjust the ROI for the target value calculation using the arrow buttons.
6. Click on the 'Calculate Target' button to calculate the target pixel value for the current ROI and method selections.

Download the Normalization Tables

The download process requires an Ethernet connection to the camera from the host computer. The direct connection requires a crossover Ethernet cable. The IP address for the camera is 192.168.1.2. Set the IP address for the computer to 192.168.1.1.

1. Exit the Normalization Calibration window and return to the MEGAPLUS Central main window.
2. From the main toolbar select File > Download > Normalization Tables.
3. From the Download Normalization Tables, click on the 'Browse' button to select the tables to download.
4. Click on the 'Download' button to automatically ftp the files to the Camera Controller.
5. When the systems confirms the download was successful, click the OK button.
6. Reset the Camera Controller. The Camera Controller will identify the new table and load it for use with the camera head(s).

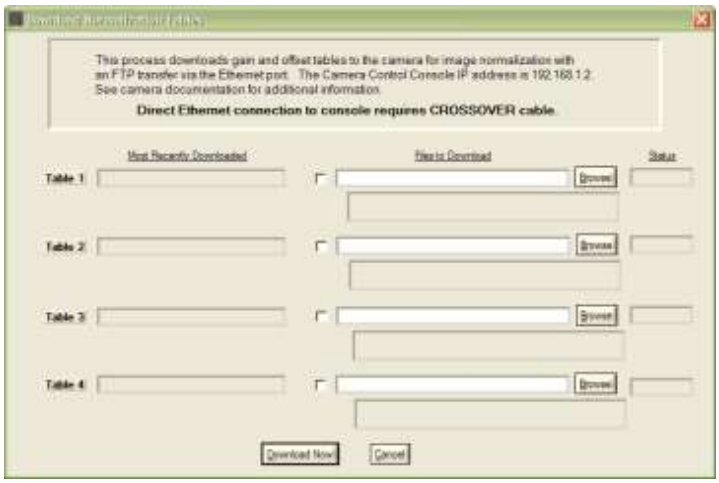

This process downloads gain and offset tables to the camera for image normalization with an FTP transfer via the Ethernet port. The Camera Control Console IP address is 192.168.1.2.
See camera documentation for additional information.
Direct Ethernet connection to console requires CROSSOVER cable.
Most Recently Downloaded
Files to Download
Status
Table 1
Table 2
Table 3
Table 4
Browse
Download Now
Cancel

Figure 9.10 Download Normalization Tables Dialog Box

Apply Flat Field Normalization

1. From the Camera Control Console, click on the 'Settings' button.
2. Click on the 'Basic' tab.
3. Click on the 'Normalization' button to apply the table. The image should be uniform.

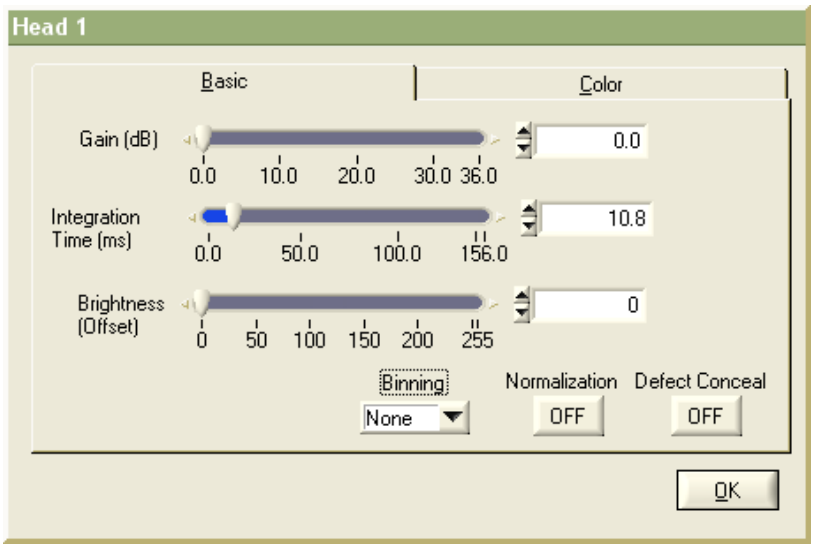


Figure 9.11 Apply Normalization Dialog Box

Histogram and Profile Display

The Histogram and Profile display shows captured or loaded reference images.

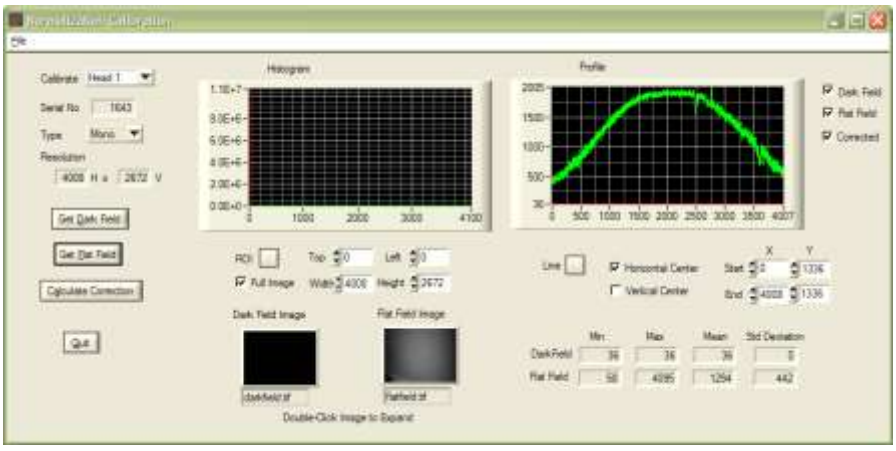


Figure 9.12 Histogram and Profile display (Normalization and Calibration window)

1. Adjust the ROI for the histogram display by using the up and down arrows to change the coordinates. By default, the histogram calculates the full image. The histogram allows evaluation of the distribution of pixel values in the reference image.
2. Adjust the profile line by using the up and down arrows to change the coordinates. The Profile window displays the horizontal center of the image. The Profile allows the evaluation of the distribution of pixel values in the reference image.

9.2.4 Flat Field Normalization with FireWire (IEEE 1394a)

MEGAPLUS Central allows image capture via FireWire interface only.

Before You Begin

- Install the NI-IMAQ IEEE 1394 Drivers.
- Install MEGAPLUS Pro version 1.35 or higher.
- In the control software, set the bit depth to 12-bits (if using raw Bayer, white balance the image).
- In the control software, set the sensor readout and clock speed.
- Ensure that the camera head is capturing images.

Overview of Steps

- Capture a dark field reference image.
- Capture a flat field reference image.
- Load the dark field and flat field reference images into the Normalization Calibration window.
- Calculate the target pixel value.
- Analyze the reference images to calculate the gain and offset values.
- Download the table to the camera.
- Reset the controller to load the new tables.
- Turn correction on or off for each Camera Head.

Figure 9.13 shows the Normalization Calibration window from which Flat Field Normalization procedures are done.

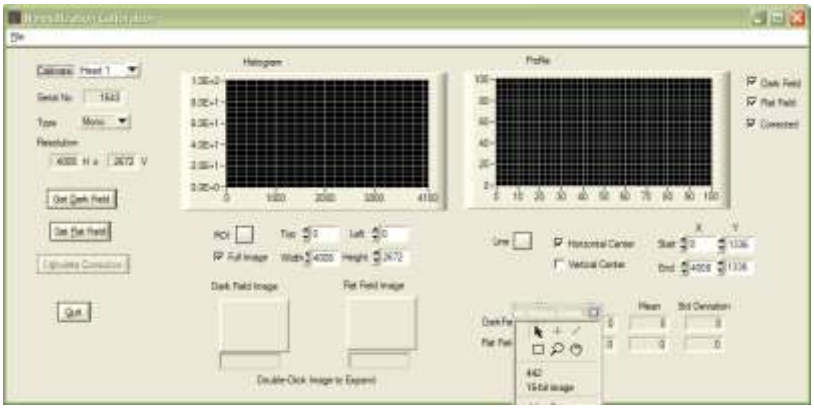


Figure 9.13 Normalization Calibration Window

Capture a Dark Field Reference Image

1. From the main toolbar, select the 'Tools' drop-down menu > select Normalization.
2. From the Normalization Calibration window, use the 'Calibrate' drop-down list to select the Camera Head to calibrate. Make sure the Camera Head is connected to the correct port. The Camera Head serial number, resolution, and type display in the Camera Head data section of the window.
3. Click on the 'Get Dark Field' button.
4. Place the lens cap on the camera head.
5. From the 'Get Dark Field Image' dialog box, select 'Capture Image' from the drop-down list.
6. If desired, change the gain and integration time using the arrow buttons.
7. Check the 'Show Saturation' check box to indicate any pixels that may be saturated.
8. Check the 'Average' check box to process the average specified number of images to create the reference image.
9. Click on the 'Acquire' button. A snapshot of the image will display in the 'Dark Field Image' window.
10. Once you have acquired a satisfactory image, click on the 'Save Image' button. MEGAPLUS Pro saves an average of the images captured. Give the file a meaningful name that includes the serial number and select the file format.
11. Click on the 'Use for Calibration' button.

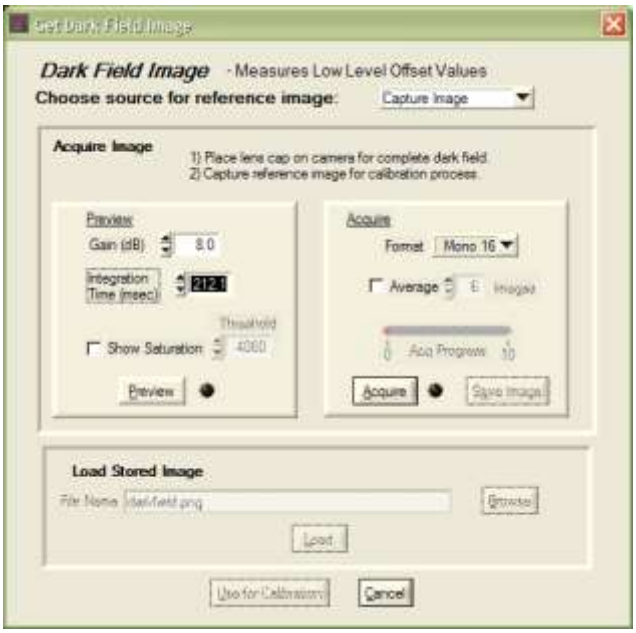


Figure 9.14 Get Dark Field Image Dialog Box

Capture a Flat Field Reference Image

1. Use the same lens, f-stop setting and external light source that will be used for the application.
2. Uniformly illuminate the target.
3. Adjust the light intensity, lens aperture and the integration time to produce an average pixel value around 80 percent of full scale.
4. From the Normalization Calibration window, click on the 'Get Flat Field' button.
5. From the 'Get Flat Field Image' dialog box, adjust the Integration time to near saturation using the arrows. Create an image that is close to saturation but not completely saturated.
6. Check the 'Show Saturation' check box.
7. The image in the 'Flat Field Image' preview window should be black with no saturated pixels.
8. Check the 'Average' check box.
9. Click on the 'Acquire' button.
10. Click on the 'Save' button. Type the file name and extension.
11. Click on the 'Use for Calibration' button.

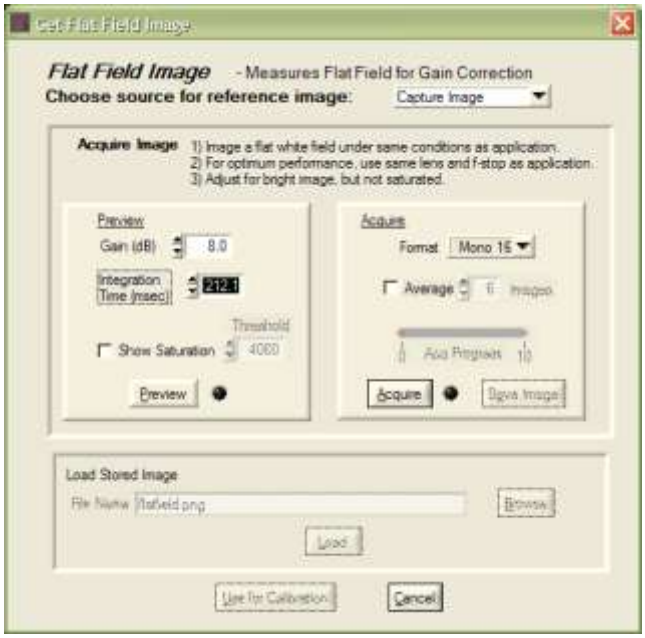


Figure 9.15 Get Flat Field Image Dialog Box

Histogram and Profile Display

The Histogram and Profile display shows captured or loaded reference images.

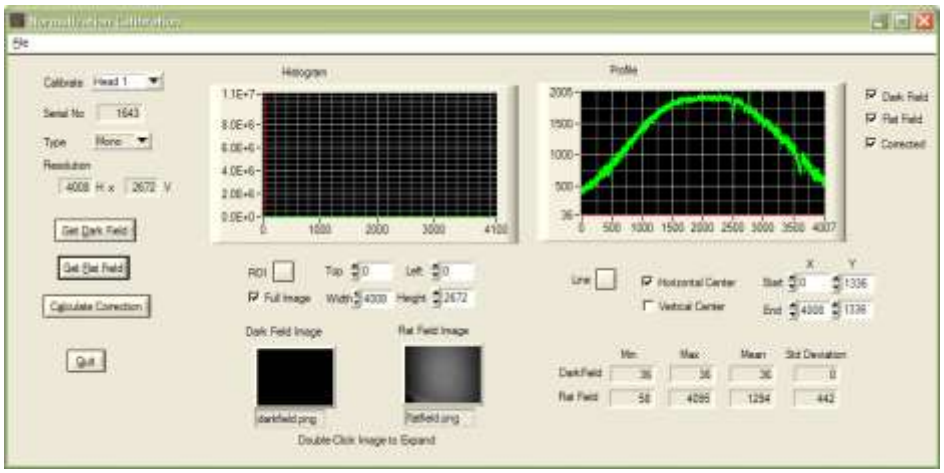


Figure 9.16 Histogram and Profile Display (Normalization Calibration Window)

1. Adjust the ROI for the histogram display by using the up and down arrows to change the coordinates. By default, the histogram calculates the full image. The histogram allows evaluation of the distribution of pixel values in the reference image.
2. Adjust the profile line by using the up and down arrows to change the coordinates. The Profile window displays the horizontal center of the image. The Profile allows the evaluation of the distribution of pixel values in the reference image.

Calculate the Correction

1. On the Normalization Calibration window, click on the 'Calculate Correction' button.
2. On the Calculate Correction dialog box, type a file name and location in the 'Output File Name' field, The extension must be .ffn.
3. If desired, type in a comment up to 80 characters in the 'Comment for Output File'. At the 'Applies To' drop-down list, select a specific Camera Head serial number to use exclusively with the new table or select the 'Any Serial No' option to use the table with any Camera Head.
4. At the 'Calculate Target With' drop-down list, select the method to derive the target value from the following:
   - Max Pixel Value—the target value will be the maximum pixel value in the ROI of the flat field image.
   - Mean Pixel Value—the target value will be the mean of the pixel values in the ROI of the flat field image.
   - Median Filter and Max—with this method, a median filter is first applied to the ROI in order to smooth the data, and then the maximum pixel value of the resulting data is selected as the target value.
   - Manual Target Value Entry—you may also choose to manually type in the target pixel value.
5. If desired, adjust the ROI for the target value calculation using the arrow buttons.
6. Click on the 'Calculate Target' button to calculate the target pixel value for the current ROI and method selections.

Figure 9.17 Calculate Correction Dialog Box

Download the Normalization Tables

The download process requires an Ethernet connection to the camera from the host computer. The direct connection requires a crossover Ethernet cable. The IP address for the camera is 192.168.1.2. Set the IP address for the computer to 192.168.1.1.

1. Exit the Normalization Calibration window and return to the MEGAPLUS Central main window.
2. From the main toolbar select File > Download > Normalization Tables.
3. From the Download Normalization Tables, click on the Browse button to select the tables to download.
4. Click on the Download button to automatically ftp the files to the Camera Controller.
5. When the systems confirms the download was successful, click the OK button.
6. Reset the Camera Controller. The Camera Controller will identify the new table and load it for use with the camera head(s).

This process downloads gain and offset tables to the camera for image normalization with an FTP transfer via the Ethernet port. The Camera Control Console IP address is 192.168.1.2.
See camera documentation for additional information.
Direct Ethernet connection to console requires CROSSOVER cable.
Most Recently Downloaded
Files to Download
Status
Table 1
Table 2
Table 3
Table 4
Browse
Download Now
Cancel

Figure 9.18 Download Normalization Tables Dialog Box

Apply Flat Field Normalization

1. From the Camera Control Console, click on the Settings button.
2. Click on the Basic tab.
3. Click on the Normalization button to apply the table. The image should be uniform.

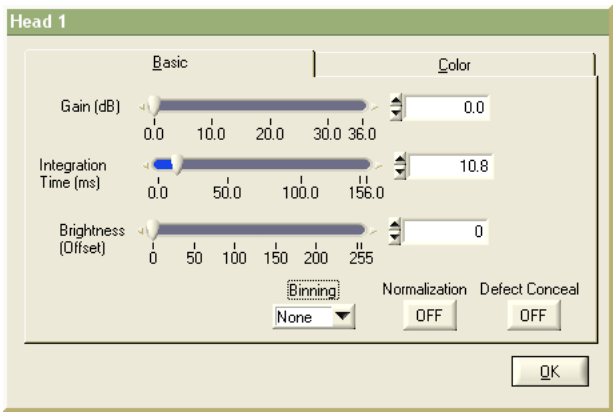


Figure 9.19 Apply Normalization Dialog Box

9.2.5 Data File Format Definition

The flat field normalization data table file includes a header section and a data section. The file extension must be "ffn". Use any file name when the files are stored on the host computer. However, when downloaded to the camera, the file name must be ffn1.ffn, ffn2.ffn, ffn3.ffn, or ffn4.ffn.

As the Camera Controller boots, it compares the serial number of each Camera Head attached, to the Camera Head serial number stored in each of the four .ffn files. It then matches and loads the normalization data with the correct Camera Head.

NOTE: All values are in the data file must be stored in a "Big Endian" format.

File Header Format
Every header field except the user data field is encoded as a four byte unsigned integer.

- Header Size - number of bytes in the header. Value = 128 bytes.
- File Type - identifies file as being flat field normalization data. Value = 0x4d31666e.
- Format Version - identifies particular format of the header and pixel data. Value = 4.
- Expected Size - number of bytes in the file, including header and data
- SensorType - identifies type of camera head served. Values as follows:

ES/EC/EP/EM11000	1
ES2001/2020	2
ES3200	3
ES4020/4010	4
ES1602/1603	6
TX285	8
ES2093	9
EC/EP/TEM16000	10

- SensorIsColor - identifies spectral type of camera head served. Value = 1 => color, Value = 0 => mono
- SerialNumber - identifies particular camera head served.
- PixelsPerLine - Values as follows:

EC/EP/TEM16000	4872
ES/EC/EP/EM11000	4008
ES2001/2020	1600
ES3200	2184
ES4020/4010	2048
ES1602/1603	1536
TX285	1004
ES2093	1920

- LinesPerFrame - Values as follows:

EC/EP/TEM16000	4872
ES/EC/EP/EM11000	2672
ES2001/2020	1200
ES3200	1472
ES4020/4010	2048
ES1602/1603	1024
TX285	1002
ES2093	1080

- BitsPerPixel - Value = 32
- UserDataSize - number of bytes in the immediately following field. Value = 80
- UserData - 80 bytes. Princeton Instruments tools that generate this file use this field for an identifying comment, but this data is not processed at all in the Camera Controller and can contain anything, in any binary or ASCII format.
- "CheckSum - sum of all bytes in the header except the checksum itself.

Example header

```
00000000  00 00 00 80 4d 31 66 6e 00 00 00 04 00 80 00 80 ....M1fn........
00000010  00 00 00 04 00 00 00 01 ff ff ff ff 00 00 08 00 ........ÿÿÿÿ....
00000020  00 00 08 00 00 00 00 20 00 00 00 50 55 6e 69 74 ....... ...PUnit
00000030  79 20 66 6f 72 20 61 6e 79 20 4b 41 49 2d 34 30 y for any KAI-40
00000040  32 30 20 63 6f 6c 6f 72 20 73 65 6e 73 6f 72 00 20 color sensor.
00000050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
00000060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ................
00000070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 13 20 ...............
```

Header data structure pseudo-code

```
#define kUserDataSize 80
struct FlatFieldNormalizationFileHdr
{
   UInt32 headerSize;
   UInt32 fileType;
   UInt32 formatVersion;
   UInt32 expectedSize;
   UInt32 sensorType;
   UInt32 sensorIsColor;
   UInt32 serialNumber;
   UInt32 pixelsPerLine;
   UInt32 linesPerFrame;
   UInt32 bitsPerPixel;
   UInt32 userDataSize;
   UInt32 userData[kUserDataSize / sizeof(UInt32)];
   UInt32 checksum;
};
```

Checksum calculation pseudo-code

```
UInt32 checksum(unsigned char * pData, size_t numBytes)
{
   UInt32 sum = 0;
   while (numBytes > 0)
   {
      sum += *pData;
      pData++;
      numBytes--;
   }
   return sum;
}
```

File Data Format

The data section of the file is an array of 16-bit words, one for each pixel location. Data is ordered for pixels starting from the upper left pixel of the image (location 0,0) proceeding left to right, top to bottom. Each word encodes the gain and offset values in the following format.

Bit 0-11 = Gain value

Bit 12-14 = Offset

Bit 15 = Used by pixel defect table

The gain value is encoded as follows:

Encoded Value (2 bytes, unsigned) = (Decimal Gain Value - 1.0) * 0x400

9.3 Pixel Defect Concealment

Pixel Defect Correction minimizes the effects of dead pixels or pixels that are too bright by correcting or concealing the problem pixels . To learn more about identifying defective pixels and outputting pixel defect concealment data files, contact a Princeton Instruments representative. The standard version of *MEGAPLUS* Central allows the activation and download of previously created defect concealment tables.

9.3.1 How to Download a Defect Concealment Table to the Camera Controller

This procedure assumes that the defect concealment tables have already been created. Defect concealment tables are specific to each Camera Head.

1. Connect the Camera Controller to the host computer using an Ethernet cable. The IP address on the Camera Controller is 192.168.1.2 and the subnet mask is 255.255.255.0. Set the IP address for the computer to 192.168.1.1.
2. From the main toolbar select File > Download > Defect Tables.
3. From the Download Defect Tables, click on the Browse button to select the tables to download.
4. Click on the Download button to automatically FTP the files to the Camera Controller.
5. When the system confirms the download was successful, click the OK button.
6. Reset the Camera Controller. The Camera Controller will identify the new table and load it.

9.3.2 How to Turn on a Previously Downloaded Defect Concealment Table

1. Find the desired camera in the Camera Control Controller and click on the Settings button below the camera icon.
2. Click on the Basic tab.
3. Click on the Defect Concealment button.

9.4 White Balance

9.4.1 Overview

MEGAPLUS Central offers both a manual and a semi-automatic color balance function for color cameras. White balancing a camera requires the use of an imaging target that contains a white field that the camera can use as a reference for which areas of an image should be white. *MEGAPLUS* Central's automatic white balance uses a statistical average of red, green and blue pixel values in the central region of an image for white balance.

9.4.2 How to Set the White Balance

1. Find the desired camera in the Camera Control Controller and click on the Settings button below the camera icon.
2. Click on the Color tab. The white balance can be set using the automatic balance feature or it can be set manually.

To set the white balance with the semi-automatic balance feature

1. Point the camera at a white target.
2. Click on the Auto Balance button. The camera automatically adjusts the red and blue channel gains up or down until a white balance is achieved based on a statistical average of RGB pixel values in the central region of an image.

To manually set the white balance

1. Point the camera at a white target and observe the resulting image.
2. Adjust the white balance by changing the blue and red channel gains up or down as required. As an alternative you can use image measurements for color values and adjust gains until the red, green and blue channels are matched.

9.4.3 How to Set Low Level Color Gains

Another option for white balance is the Low Level Color Gains adjustment. This function controls the low-level analog color gains applied to individual pixels in the camera head's signal processing circuitry for a Bayer pattern sensor. There is a separate gain control function for red, green and blue.

1. From the *MEGAPLUS* Control Controller, click on the Settings button.
2. Click on the Color tab.
3. Use the Low Level Color Gain arrow buttons to adjust each color separately.
4. To reset the values to zero, click on the Zero Color Gains button.

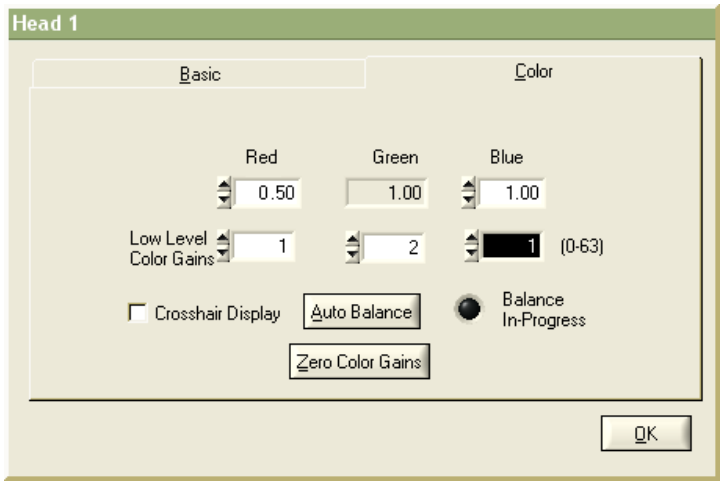


Figure 9.20 Camera Head Control Window- Color Tab

9.4.4 White Balance Technical Points

White balance is the adjustment of the relative amplitude of the signal from each color plane of the image. A color camera that is not properly white balanced will create an image in which the colors are incorrect, often creating a green or red tinge to objects that should appear white. Balancing is usually implemented by imaging a white surface and then making adjustments until the digital number for the pixel data in each color plane is approximately equal. This is done because in electronic imaging systems, the color white is created by displaying equal amounts of red, green, and blue. If the system is adjusted such that white is displayed properly, then other colors will be displayed accurately. Low Level Color Gains must be set to at least 1 for White Balance to work.

9.5 How to Display the Crosshairs for Color Cameras

1. Find the desired camera in the Camera Control Controller and click on the Settings button below the camera icon. Click on the Color tab.
2. Check the Crosshair Display box.

NOTE: The crosshair display is for color cameras only. Also, GigE Vision does not support crosshairs.

9.6 How to Set the Trigger Parameters

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Trigger/Strobe tab to access the Trigger Mode, Polarity and Source functions.

9.6.1 How to Set the Trigger Mode

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Trigger/Strobe tab to access the Trigger Mode functions.
3. Use the pull-down list to select from the following modes:

Mode Selection	Result
Trigger Off	Free run, continuous video
Mode 0	Edge Triggering (Asynchronous Reset with Programmable Integration
Mode 1	Level Triggering (Asynchronous Reset with Pulse Width Controlled Integration
Mode 4	Double Exposure Triggering
Mode 6	Periodic Interval (Self Trigger)
Mode 7	Overlap Triggering (Asynchronous Reset with Programmable Integration and Readout During Integration)

Table 9.1 Trigger Mode Selection

For more information, see "Operation Modes: Triggering or Continuous Video" on page 77.

9.6.2 How to Set the Trigger Source

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Trigger/Strobe tab to access the Trigger Mode, Polarity and Source functions.
3. Use the Source pull-down menu to select from the following options:
    - BNC
    - CameraLink
    - Software

9.7 How to Set the Strobe Parameters

A delay can be added to the strobe output pulse. The time delay starts when the trigger is received and is delayed for the duration set in the Delay field.

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Trigger/Strobe tab to access the Polarity and Delay functions.

9.7.1 How to Set the Strobe Output Polarity

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Trigger/Strobe tab to access the Polarity functions.
3. Use the pull-down menu to select either negative or positive polarity.

9.7.2 How to Set the Strobe Output Delay

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Trigger/Strobe tab to access the Delay functions.
3. Use the up and down arrow keys to enter a value or type the value directly into the Delay field.

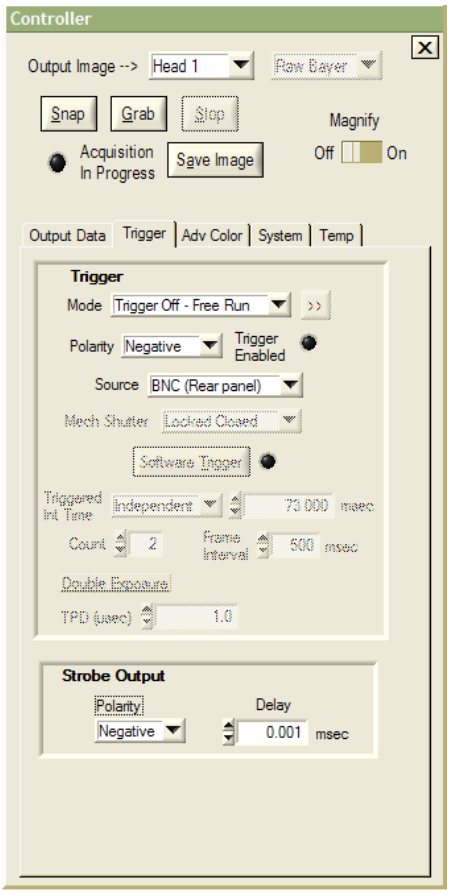


Figure 9.21 Trigger and Strobe Settings

9.8 Color Processing

9.8.1 How to Set the Gamma Control

The output Look Up Tables (LUTs) are configured by the Gamma Control. When Gamma is turned on and a value other than one is selected then the appropriate gamma curve is calculated in the controller.

Activating the automatic gamma control:

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Adv Color tab to access the gamma settings.
3. Click the Gamma button to the ON position.

9.8.2 How to Set the Color Space Conversion

When the Color Space Conversion Engine in turned on, the input RGB Look Up Tables (LUTS), the 3 x 3 matrix and the RGB output tables are active. In the OFF position, Look Up Tables are loaded into the input LUTs, a 3 x 3 matrix is loaded with 1s on the diagonal and 0s in the other cells.

Activating the color space conversion engine:

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Adv Color tab to access the gamma settings.
3. Select the pre-configured Coefficient Set from the pull-down menu.
4. Click the Color Space Conversion button to the ON position. The Input and output RGB lookup tables and the 3 x 3 matrix are activated.

NOTE: Color Processing functions apply to the selected head only. Also, GigE Vision does not currently support color space conversion.

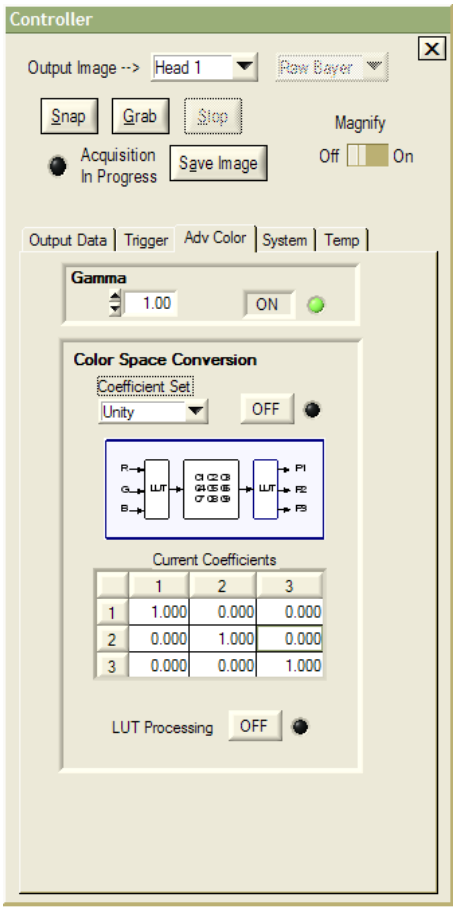


Figure 9.22 Controller Control Gamma and Color Space Conversion

For more information: See "Color Space Correction" on page 201.

9.9 Temperature Control

The controls in the Auto Temp panel, allow you to set the target temperature for the active camera(s) and activate/deactivate auto temperature control. The target temperature setting will only be used if Auto is turned on by clicking on the Hd# button.

Depending on the camera, passive cooling and active cooling (TEC and/or fan) may be available. The TEC and Fan switches allow you to activate active cooling. When Auto Temp is ON, the TEC On/Off setting is ignored.

The information in the Temperature (Celsius) panel reports the current temperatures for the sensor, camera (head), and console (controller). The three temperatures reported by Console (controller) temperature are the temperature at the top of the controller processor board, at the bottom of the controller processor board, and the temperature of the controller power supply board.

Clicking on the **Refresh** button retrieves the latest temperature information.

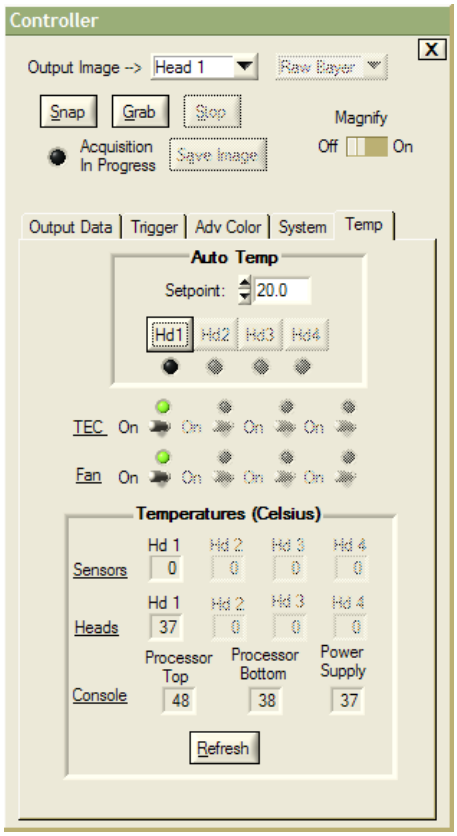


Figure 9.6 Temperature Control

10. Save and Restore Camera Settings

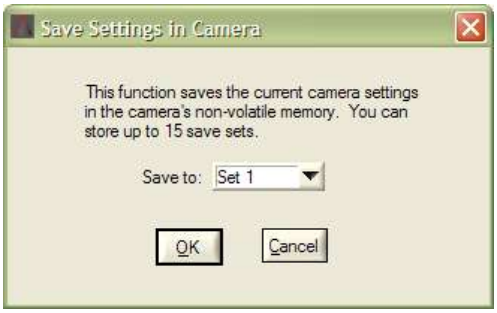
The *MEGAPLUS* platform supports saving the state of the camera's user-controlled operating parameters and restoring them on demand. This capability is supported directly from the GUI, the API and the Control Protocol. There are a maximum of 14 user-definable camera sets that can be saved to the camera's compact flash. The camera keeps track of the set number that was specified the last time settings were saved. When the camera is powered down and then powered up, it will load the last specified saved settings. If a set is not saved, it will load the factory defaults.

Camera Settings Stored with Each Saved Set
Edge Mode (Integrate and Dump Independent) Level Mode (Integrate and Dump Level) Double Exposure Asynchronous Reset Periodic Interval Edge-Overlap
Trigger enable - Off/On Trigger polarity - Positive / Negative Trigger source - BNC / CameraLink / Software Trigger count - Numeric value for counted modes Triggered Integration Time - Numeric value Mode 6 Self Trigger Interval - Numeric interval value for periodic interval mode Double Expo TPD - Transfer Pulse Delay value for double exposure mode
Strobe Polarity - Positive / Negative Strobe Delay - Numeric value for delay interval
Gamma - Gamma processing enable, On/Off Gamma value - Floating point numerical value for gamma
Color Space processing enable - On/Off Color Space Coefficient 1 - Floating point values from 0.0 to 2.0 for all 9. Color Space Coefficient 2 through 9 Color space Lookup Table processing enable - On/Off Tables are presumed to be loaded onto file system.
Output Bit Window Bit Depth - 8/10/12 Bit window least significant bit - 0,1,2,3,4
Tap Readout - Single/Dual Tap
CameraLink Mux
CameraLink Port A through F input sensor head 1,2,3, or 4
CameraLink Config Class - Medium/Dual/Alt Tap
QuickMux Value - sensor head 1,2,3,4, or "custom" (99) Defaults to leftmost attached head if saved head is not present.
Pixel Clock Speed - High/Low Clock frequency is derived from clock speed and sensor type.
Analog/DVI Video Output - On/Off Currently, always forced off to prevent unintentional on state.

Table 10.1 Saved Camera State Settings

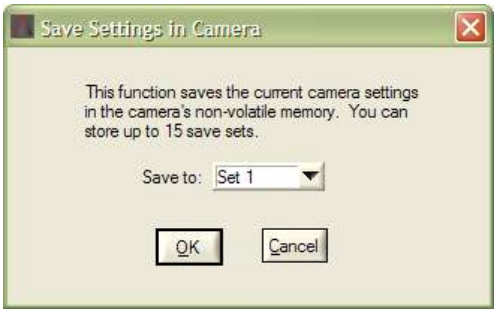
10.1 How to Save Camera State Settings

1. From the main toolbar, select the File pull-down list > Save Settings.
2. From the Save Settings Camera dialog box, use the pull-down list to select a Set (1-14).
3. Click OK to save the camera settings under the selected set.


Save Settings in Camera
This function saves the current camera settings in the camera's non-volatile memory. You can store up to 15 save sets.
Save to:
Set 1
OK
Cancel

10.2 How to Restore Camera Settings

1. From the main toolbar, select the File pull-down list > Load Settings.
2. From the Load Settings Camera dialog box, use the pull-down list to select a Set (Default-14). The Default Set is the factory default. Set numbers 1-13 are user-defined.
3. Click OK to load the camera settings.


Save Settings in Camera
This function saves the current camera settings in the camera's non-volatile memory. You can store up to 15 save sets.
Save to:
Set 1
OK
Cancel

NOTE: Camera settings are not saved to individual camera heads. If a Camera Head is moved to a different Camera Controller, the Camera Head acquires the settings of the new Camera Controller.

11. Camera Head Configuration Settings

Applies to multi-head controller only.

11.1 How to Select DVI Output

The DVI outputs only one selected Camera Head view at a time. To view the output from a different Camera Head, use the Output Image pull-down list at the top of the control panel.

1. Select the desired camera from the Output Image pull-down list at the top of the Control panel.
2. Click on the System tab to access the Configuration Settings.
3. Click on the DVI Output button.

See "Camera Controller Front and Back Panel Connectors" on page 17.

WARNING!: DO NOT use the DVI output option for extended periods on Camera Controllers that are not equipped with an integrated cooling fan. Using the DVI output option generates significant heat inside the Camera

11.2 How to Change the Camera Configuration Settings

1. Select the desired camera from the Output Image pull-down list at the top of the Control panel.
2. Click on the System tab to access the Configuration Settings.
3. Select the desired FPGA from the Configuration Loaded on Next Reboot pull down.
4. Click on the Reset Controller button to load the new selection.

11.3 How to Reset the FireWire IEEE 1394a bus

IEEE 1394a applies to multi-head controller only.

The FireWire IEEE 1394a bus can be reset from *MEGAPLUS* Central allowing the equivalent of unplugging and plugging in the FireWire cable.

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the System tab to access the Configuration Settings.
3. Click on the 1394 Bus Reset button.

Figure 11.1 Control Panel - System Tab

11.4 Camera Control and Image Acquisition Configuration

MEGAPLUS cameras can be controlled via FireWire, Ethernet, the serial port in CameraLink, or a 9-pin RS-232 serial port.

11.4.1 How to Set Camera Control and Image Acquisition Options

Camera Control	Image Acquisition	From the System Camera List Select
FireWire	FireWire	FireWire (1394)
Serial RS-232	FireWire	FireWire (1394) with Serial Control
CameraLink	CameraLink frame grabber	CameraLink Serial, and select the Vendor's DLL from the pull-down list
Virtual COM Port		COM Port Serial, and select the virtual COM port number from the COM Port pull-down list
GigE Vision	Gigabit Ethernet	Gigabit Ethernet

11.4.2 How to Change the Interface

1. From the File menu on the toolbar select Camera List.
2. Select the camera from the Installed Camera List.
3. Select the desired control interface and then click OK.

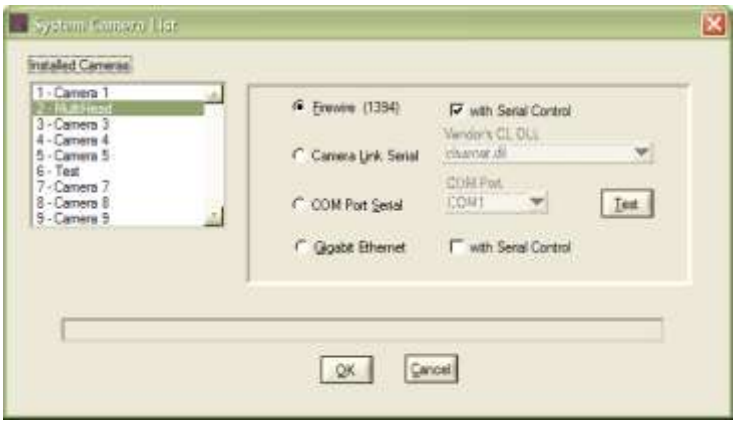


Figure 11.2 System Camera List - Camera Control

For more information:

See "Camera Control and Image Acquisition Interfaces" on page 9.

This page intentionally left blank.

12. Image Display and Acquisition

12.1 FireWire Image Acquisition

Applies to multi-head controller only.

To acquire images via FireWire, select FireWire camera control and image acquisition as previously described in the section "Camera Control and Image Acquisition Interfaces" on page 9. The Controller Control panel will automatically be configured to reflect the FireWire image acquisition options.

12.1.1 How to Set Continuous Image Acquisition

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Grab button to open the image display window and start the image acquisition.
3. Click on the Stop button to complete the image acquisition.
4. Click on the Save Image button to open the Save dialog. Images can be saved to the hard disk or other storage device in the following formats: TIFF, JPEG, PNG or RAW.

NOTE: Image data is not stored in the camera during continuous capture when using a FireWire interface. Only the last displayed image will be saved.

12.1.2 How to Capture a Single Image

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Snap button to open the image display window and acquire a single image.
3. Click on the Save Image button to open the Save Dialog. Images can be saved to the hard disk or other storage device in the following formats: TIFF, JPEG, PNG or RAW.

12.1.3 How to Scale (Zoom/Unzoom) the Visible Image

Images can be scaled using the Zoom setting on the Controller Control panel. Generally a Zoom level of 3 to 7 produces an image that is smaller than the active viewing area. For viewing larger images, click on the Full Screen button in the bar along the top of the image viewing window to resize the display area. To zoom out on an image, select the Zoom tool from the Controller Control panel and hold down the Shift key and right-click the mouse.

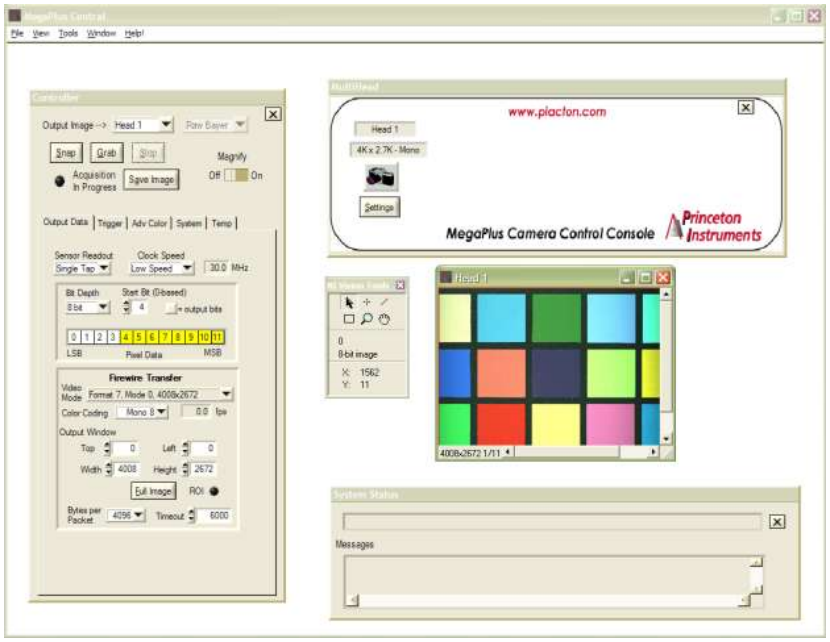


Figure 12.1 Image Display

12.1.4 How to Select the Region of Interest

The Region of Interest (ROI) feature creates a rectangular-shaped subset of the entire image. ***This feature is used exclusively with FireWire configurations.*** An ROI of any size may be selected; however, the resulting image will be determined by the constraints of the camera. The ROI feature is ideal for the following situations:

- When only a small section of the image needs to be analyzed.
- To increase acquisition speed.
- To decrease the amount of bandwidth required for moving several files simultaneously.

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Select the Output Data tab.
3. Click on the Snap button to open the image display window and acquire a single full image. Consider saving this full image as a reference to check the position of the ROI.
4. Under the Output Window settings, use the Width, Height, Top and Left arrow keys to select the ROI or type the ROI coordinates in the text boxes. The image coordinates display in the IMAQ Tools box.
5. Once the desired ROI is selected, click on the Save Image button.

NOTE: If a full image is required to reference the exact position of the ROI image, capture and save a full image or use the DVI output.

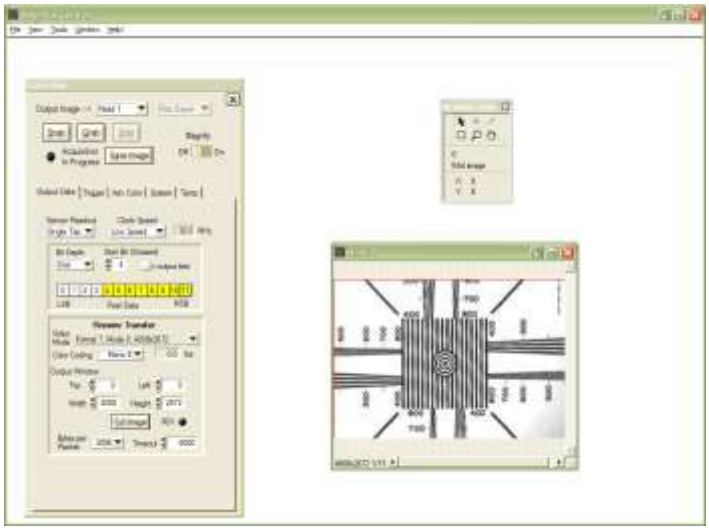


Figure 12.2 ROI Dialog Box

12.1.5 How to Set the Bit Windowing

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Select the Output Data tab.
3. Select the Output Bit Depth from the Bit Depth pull-down menu.
4. Select the Start Bit by clicking on the up or down arrow keys.

For more information:

See "Bit Windowing Overview" on page 199.

12.2 FireWire and *MEGAPLUS* Image Bit Depths and Image Formats

FireWire supports specific image bit depths and specific image formats. It supports other image formats via Format 7, which allows specification of the image format. FireWire supports 24- and 48-bit RGB, 8- and 16-bit monochrome. If a 12-bit mono image is acquired and saved with FireWire, the image is saved in a 16-bit format. *MEGAPLUS* Central supports RGB images with bit depths up to 8-bits.

NOTE: Image output of 48-bit RGB, requires a camera driver other than the NI-IMAQ IEEE 1394a Drivers.

12.3 How to Configure the Transfer Format Using FireWire

MEGAPLUS Central allows changes to the standard camera to PC transfer settings. In the Video Mode pull-down menu, the Scalable Image, Mode 0 is used for virtually all *MEGAPLUS* Camera Heads because the Camera Head resolutions do no match the pre-defined image resolutions in the FireWire Standard (IIDC 1394-based Digital Camera Specification Version 1.31. by the 1394 Trade Association). When Format 7, Scalable Image, Mode 0 is selected, the camera automatically configures the output resolution of the image. The output resolution is displayed in the Output Window parameters.

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Select the Output Data tab.
3. In the FireWire Transfer section, select the Video Mode.

4. Select the color coding method for image display and saving images from the pull-down menu. For color heads, this should be set to 24-bit RGB. For mono heads, this can be set to 8-bit or 16-bit mono.

12.4 GigE Vision Configuration and Output

Applies to GigE Vision single-head controller only.

To acquire images via GigE Vision, select Gigabit Ethernet camera control and image acquisition as previously described in the section "Camera Control and Image Acquisition Interfaces" on page 9. The Controller Control panel will automatically be configured to reflect the GigE Vision image acquisition options.

12.4.1 How to Set Continuous Image Acquisition

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Grab button to open the image display window and start the image acquisition.
3. Click on the Stop button to complete the image acquisition.
4. Click on the Save Image button to open the Save dialog. Images can be saved to the hard disk or other storage device in the following formats: TIFF, JPEG, PNG or RAW.

NOTE: Image data is not stored in the camera during continuous capture when using a GigE Vision interface. Only the last displayed image will be saved.

12.4.2 How to Capture a Single Image

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Click on the Snap button to open the image display window and acquire a single image.
3. Click on the Save Image button to open the Save Dialog. Images can be saved to the hard disk or other storage device in the following formats: TIFF, JPEG, PNG or RAW.

12.4.3 How to Scale (Zoom/Unzoom) the Visible Image

Images can be scaled using the Zoom setting on the Controller Control panel. Generally a Zoom level of 3 to 7 produces an image that is smaller than the active viewing area. For viewing larger images, click on the Full Screen button in the bar along the top of the image viewing window to resize the display area. To zoom out on an image, select the Zoom tool from the Controller Control panel and hold down the Shift key and right-click the mouse.

Figure 12.3 Image Display

12.4.4 How to Set the Bit Windowing

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Select the Output Data tab.
3. Select the Output Bit Depth from the Bit Depth pull-down menu.
4. Select the Start Bit by clicking on the up or down arrow keys.

For more information:

See "Bit Windowing Overview" on page 199.

12.5 CameraLink Configurations and Output

12.5.1 How to Select Data Output Options

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Select the Output Data tab.
3. From the Sensor Readout pull-down list, select from the following options
   - Dual Tap
   - Single Tap
4. From the Clock Speed pull-down list, select from high or low speed.
5. From the Bit depth pull-down list, select the desired color or monochrome bit depth for the camera head.
6. Select the Start Bit.

12.5.2 How to Select the CameraLink Configuration

The CameraLink interface can be used in Base, Medium or Dual output formats. *MEGAPLUS* also supports alternating tap. Alternating tap divides the output image pixel stream into two parallel streams, one consisting of the odd numbered pixels and the other consisting of the even numbered pixels. This allows operating camera heads at their maximum frame rate at half the pixel clock rate required for single tap output.

1. Select the desired camera from the Output Image pull-down list at the top of the Controller Control panel.
2. Select the Output Data tab.
3. From the CameraLink Cfg pull-down list, select from the following options
    - Base/Medium
    - Dual Base [*]
    - Dual Base, Alternating Tap

12.6 How to Set Multi-head Configurations with CameraLink

Make sure that the system is running in Four Head Mono Mode.

1. Select Custom from the Output Image pull-down list at the top of the Controller Control panel.
2. Select the Output Data tab.
3. From the CameraLink Cfg pull-down list, select Base/Medium
4. From the pull-down list select Tap 1 for Head 1, Tap 2 for Head 2 etc. The image will be a compilation of both Camera Heads. Example: two ES 2001[**] heads, set the frame grabber to see one 3200 x 1200 image.

Dual Base Frame Grabber

Dual Base frame grabbers run a maximum of two Camera Heads. If using a Dual Base frame grabber:

- Set Head 1 to CameraLink 1
- Set Head 2 to CameraLink 2

For more information see "Camera Control and Image Acquisition Interfaces" on page 9.

[*] With Firmware version 1.61 or later, Single-Head Controllers support Dual Base and Dual Base, Alternating Tap.
[**] ES 1602 and ES 2001 have been discontinued.

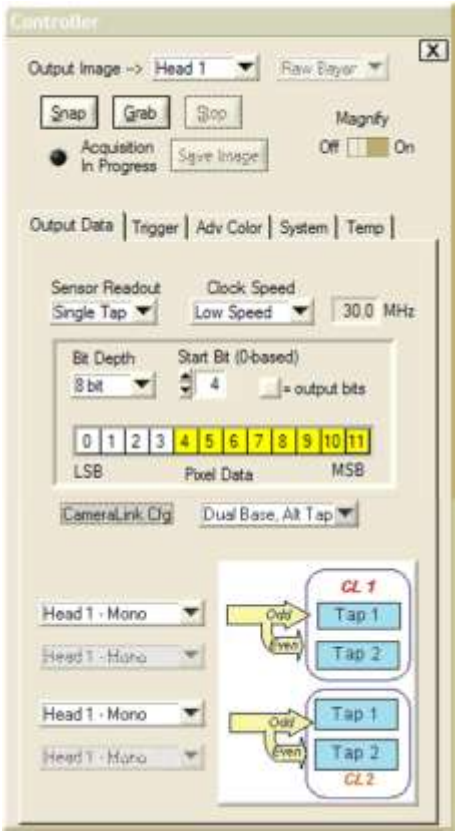


Figure 12.4 Output Data Tab - CameraLink Dual Base, Alt Tap

This page intentionally left blank.

13. Operation Modes: Triggering or Continuous Video

The trigger signal initiates the acquisition and transfer of a single frame of data in one of several possible ways. The source of the trigger input is selected via a camera control function. The source of the trigger can come from an electrical pulse or it can be generated internally within the Camera Controller. The polarity of the External Trigger signal is user programmable. The external trigger signal input may be derived from one of two sources: 1) the Trigger-In BNC connector on the rear panel or 2) the Camera Control 1 line or the CameraLink interface.

For related information see "How to Set the Trigger Parameters" on page 59.

13.1 Trigger Off; Free Run, Continuous Video

When the trigger state is set to OFF, the camera operates in Free Run or Continuous Video mode. This mode requires no external control signals and provides high frame rates by overlapping the readout time with the exposure time. An internally generated, fixed frequency trigger signal initiates the readout of the current frame and starts the exposure time for the next frame. The frame rate is controlled internally. Integration time is programmable.

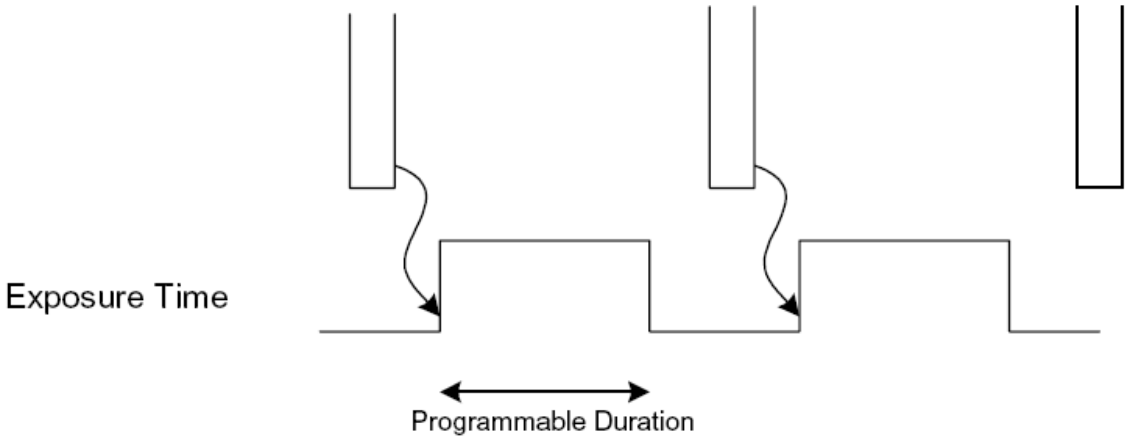


Figure 13.1 Free Run Mode - No External Trigger

13.1.1 Minimum Integration Times for Continuous Video Mode

Minimum integration times in the table were measured by setting the integration time to 0 in MEGAPLUS Central. The times were measured from the Clear pulse to the falling edge of the V3 pulse (shift charge to vertical shift registers).

Sensor	Taps	Speed	Minimum Integration Time
KAI-16000	1	30 MHz	880 µs
	2	30 MHz	840 µs
	1	38 MHz*	740 µs
	2	38 MHz*	670 µs
KAI-11002	1	30 MHz	312 µs
	2	30 MHz*	245 µs
	1	38 MHz*	246 µs
	2	38 MHz*	192 µs
KAI-4020	1	30 MHz	167 µs
	2	30 MHz	133 µs
	1	38 MHz	132 µs
	2	38 MHz	104 µs
KAI-2093	1	30 MHz	150 µs
	2	30 MHz	120 µs
	1	38 MHz	120 µs
	2	38 MHz	94 µs
KAI-2020 KAI-2001	1	30 MHz	140 µs
	2	30 MHz	114 µs
	1	38 MHz	111 µs
	2	38 MHz	90 µs

* Running the KAI-16000 or KAI-11002 in an over clocked state (38 MHz) is not recommended.

Table 13.1 Minimum Integration Times for Free Run Mode

NOTE: Full frame sensors such as the 3200 and the 1602/1603 do not perform in Free Run mode. Trigger mode must be set to ON and a trigger mode must be selected.

13.2 Trigger Modes

Triggered acquisition can be configured for a variety of different operating modes. These triggered modes provide different methods of controlling the start of image acquisition and the duration of the integration time. When the trigger state is OFF, the camera is in continuous, free run or "video mode." When trigger state is ON, the user can select from the following modes:

0 = Edge Triggering (Asynchronous Reset with Programmable Integration)

1 = Level Triggering (Asynchronous Reset with Pulse Width Controlled Integration)

4 = Double Exposure Triggering

6 = Periodic Interval Triggering (Self Trigger)

7 = Overlap Triggering (Asynchronous Reset with Programmable Integration and Readout During Integration)

13.3 Mode 0; Edge Triggering (Asynchronous Reset with Programmable Integration)

13.3.1 Overview

In this mode the active edge (first edge) of EXT TRIG initiates the start of a programmable integration time. At the end of the integration time the readout takes place. After the readout, the system is ready for another EXT TRIG signal.

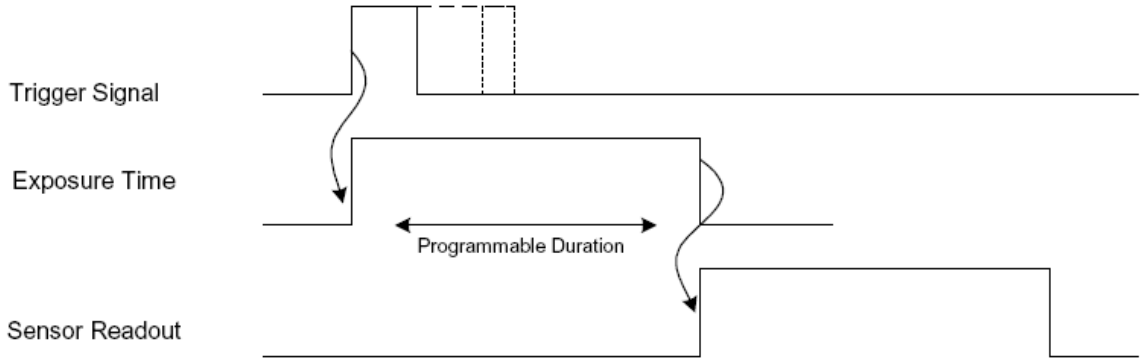


Figure 13.2 Basic Edge Trigger Mode – User-programmable Integration

13.3.2 Mode 0 Edge Triggering for Interline Sensors

In Edge mode triggering, the camera responds to a trigger edge by clearing the sensor photosites and letting the sensor integrate for a preset length of time. At the end of the integration time, the image data is read from the sensor. The minimum time between triggers is approximately equal to the integration time plus the readout time. Readout times for the various cameras can be found in Table 13.9 on page 86.

NOTE: Faster triggering is available using Mode 7. See "Mode 7; Overlap Triggering (Asynchronous Reset with Programmable Integration and Readout During Integration)" on page 74.

13.3.3 Mode 0 Timing Sequence for Interline Sensors

In this mode, the sensor is continually being flushed of charge while waiting for a trigger. When a trigger arrives via the rear panel, the photosites are cleared and a strobe is issued out of the rear panel. At the end of the CLEAR cycle, a timer is started which has been pre-programmed with the desired integration time. During the integration period the sensor continues to accumulate charge. When the integration time expires the image is read from the sensor.

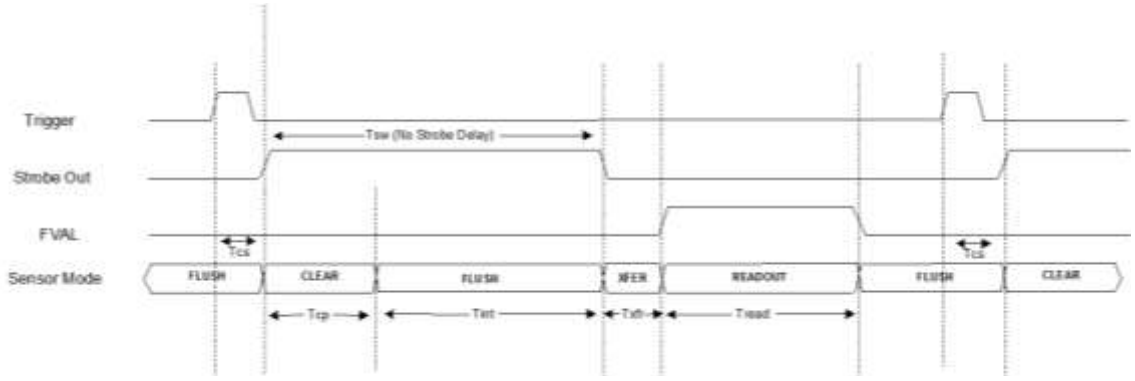


Figure 13.3 Triggering with No Strobe Delay for Interline Devices

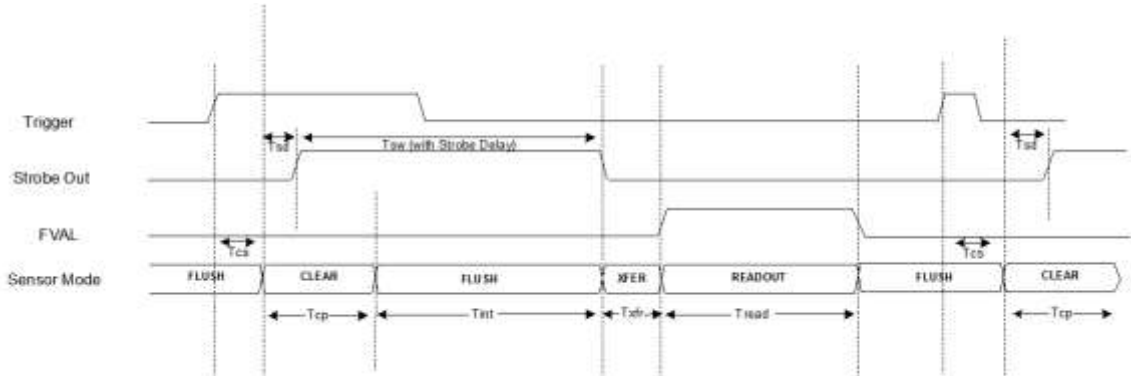


Figure 13.4 Triggering with Strobe Delay for Interline Devices

13.3.4 Mode 0 Timing Parameters for Interline Sensors

The following definitions pertain to Figure 13.3 and Figure 13.4.

Tcs (Clear Start Latency): This 2 us interval defines the delay from the receipt of trigger to the start of the clear pulse to the sensor. This is the reaction time of the optoisolator.

Tsd (Strobe Delay): This interval defines an optional, user-adjustable delay from the start of the clear pulse to the issue of the strobe pulse. By default the strobe pulse begins at the time the clear pulse ends (i.e. Tsd = 0). The delay is under software control and is programmable from 0 to 1 second in 100 ns increments.

Tsw (Strobe Width): The duration of the strobe width equals the time of the clear pulse (Tcp) pulse the integration time (Tint) minus any optional strobe delay (Tsd). By default the strobe pulse begins at the time the clear pulse is sent to the sensor. The start of the strobe pulse may be delayed by a user adjustable setting in the software.

Tcp (Clear Pulse): This is the time it takes to clear the sensor for interline sensors. The start of clear pulse begins as soon as possible after the camera receives a trigger (after Tcs). This time is sensor dependent. There is a nominal uncertainty in the start latency of +/- 1 pixel period (1/pixel clock rate.)

Sensor	Sensor Frequency	Tcp
KAI-16000	30 MHz	15 µs
KAI-11002	30 MHz	19 µs
KAI-4020	30 MHz	19 µs
KAI-4020	38 MHz	15 µs
KAI-2093	30 MHz	25.3 µs
KAI-2093	38 MHz	20 µs
KAI-2001	30 MHz	25.3 µs
KAI-2001	38 MHz	20 µs

Table 13.2 Sensor/Sensor Frequency/Tcp

Tint (Integration Time): This is a user programmable interval which can range from 10 µs to 10 seconds. For interline sensors the absolute accuracy of the integration time is approximately 1 µs with an uncertainty of +/-1 pixel period.

Sensor	Taps	Speed	Mode 0 Minimum Integration Time
KAI-16000	1	30 MHz	780 µs
	2	30 MHz	780 µs
	1	38 MHz*	640 µs
	2	38 MHz*	610 µs
KAI-11002	1	30 MHz	133.6 µs
	2	30 MHz	133.6 µs
	1	38 MHz*	105.2 µs
	2	38 MHz*	105.2 µs
KAI-4020	1	30 MHz	88.9 µs
	2	30 MHz	88.9 µs
	1	38 MHz	69.8 µs
	2	38 MHz	69.8 µs
KAI-2039	1	30 MHz	72 µs
	2	30 MHz	72 µs
	1	38 MHz	56.4 µs
	2	38 MHz	56.4 µs
KAI-2020	1	30 MHz	71.6 µs
KAI-2001	2	30 MHz	71.6 µs
	1	38 MHz	56.2 µs
	2	38 MHz	56.2 µs

* Running the KAI-16000 or KAI-11002 in an over clocked state (38 MHz) is not recommended.

Table 13.3 Mode 0 Minimum Integration Times

Txfr (Transfer Time): This interval represents the time it takes to transfer the data from the photosites to the vertical shift register. This time is sensor dependent.

Sensor	Sensor Frequency	Txfr
KAI-16000	30 MHz	600 µs
KAI-11002	30 MHz	125 µs
KAI-4020	30 MHz	114 µs
KAI-4020	38 MHz	90 µs
KAI-2093	30 MHz	95 µs
KAI-2093	38 MHz	75 µs
KAI-2001	30 MHz	95 µs
KAI-2001	38 MHz	75 µs

Table 13.4 Sensor/Sensor Frequency/Txfr

Tread (Readout Time): This interval represents the time it takes to read the image from the sensor. This time is sensor dependent. The following readout times are measured results.

Sensor	Taps	Sensor Frequency	Tread
KAI-16000	1	30 MHz	588 ms
	2	30 MHz	322 ms
	1	38 MHz*	476 ms
	2	38 MHz*	256 ms
KAI-11002	1	30 MHz	392 ms
	2	30 MHz	212 ms
	1	38 MHz*	309.5 ms
	2	38 MHz*	167.3 ms
KAI-4020	1	30 MHz	153.7 ms
	2	30 MHz	82.8 ms
	1	38 MHz	121.4 ms
	2	38 MHz	65.4 ms
KAI-2093	1	30 MHz	76.5 ms
	2	30 MHz	41.4 ms
	1	38 MHz	60.4 ms
	2	38 MHz	32.7 ms
KAI-2001	1	30 MHz	72.7 ms
	2	30 MHz	40.6 ms
	1	38 MHz	57.4 ms
	2	38 MHz	32.1 ms

* Running the KAI-16000 or KAI-11002 in an over clocked state (38 MHz) is not recommended.

Table 13.5 Sensor/Sensor Frequency/ Single Tap/Dual Tap/Tread Interline Sensors

Tr (Recovery Time): This interval represents the time from the end of sensor readout to the time the system can accept another trigger pulse. The recovery time is approximately 1 µs.

Maximum Trigger Rate: The maximum trigger rate is related to the minimum time between triggers which may be calculated as the sum of Ts + Tint + Txfr + Tread + Tr.

13.3.5 Mode 0 Timing Parameters Full Frame Sensors

In edge mode triggering, the camera responds to a trigger edge by ending the flush of the photosites, opening a shutter and letting the sensor integrate for a preset length of time. At the end of integration time the shutter closes and the image is read from the sensor. The minimum time between triggers is approximately equal to the integration time plus the readout time.

13.3.6 Mode 0 Timing Sequence for Full Frame Sensors

In this mode, the sensor is continually being flushed of charge while waiting for a trigger. When a trigger arrives via the rear panel, the photosites are cleared and a strobe is issued out of the rear panel. A timer pre-programmed timer starts with the desired integration time. During integration the sensor clocking stops. When integration time ends, the shutter closes and the image is read from the sensor following a short delay.

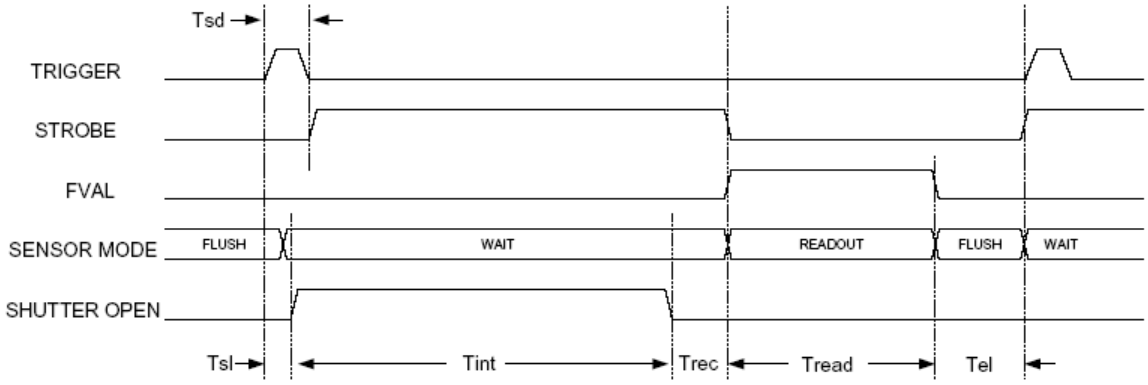


Figure 13.5 Mode 0 Timing Diagram for Full Frame Sensors

13.3.7 Mode 0 Timing Parameters for Full Frame Sensors

The following definitions pertain to Figure 13.5.

Tsd (Strobe Delay): This interval defines the delay from the receipt of a trigger to the beginning of the strobe pulse. It is a programmable delay from 0 µs to 1 second in 100 ns increments

Tsl (Start Latency): This is the time from the leading edge of the trigger pulse to the start of integration. The latency is determined by the time required to open the mechanical shutter, usually a few milliseconds.

Tint (Integration Time): This user programmable interval can range from 10 µs to 10 seconds. For full frame sensors the absolute accuracy of the integration time is determined by the mechanical response time of the shutter.

Trec (Recovery Time): This delay is between the close of the shutter and the start of the sensor readout. It allows the mechanical shutter to close before readout begins. Its value is 20 ms.

Tread (Readout Time): This interval represents the time it takes to read the image from the sensor. The time is dependent on the sensor.

Sensor	Sensor Clock Rate	Pixel Clock Rate	Data Valid Clock Rate (DVAL)	Tread
ES 3200	12 MHz	60 MHz	12 MHz	300 ms
ES 1602*/1603	12 MHz	60 MHz	12 MHz	148 ms

* ES 1602 has been discontinued.

Table 13.6 Sensor Clock Rate/Tread for Full Frame Sensors

NOTE: Data Valid must be used by the frame grabber. The pixel clock runs faster than the sensor clock.

Tel (End Latency): This interval represents the time from the end of the sensor readout unit the system can accept a subsequent trigger pulse. The end latency is approximately 1 µs.

Maximum Trigger Rate: The maximum trigger rate is dependent on the minimum time between triggers. Calculate the Trigger rate as the sum of Tsl + Tint + Trec + Tread +Tel.

Strobe Pulse Width: The leading edge of the strobe pulse occurs after the Tsd from the receipt of a trigger. The trailing edge of the strobe occurs at the start of sensor readout. The value of the user-defined Tsd must be greater than Tsl + Tint +Trec.

13.4 Mode 1; Level Triggering (Asynchronous Reset with Pulse Width Controlled Integration)

13.4.1 Overview

In this mode, both edges of EXT TRIG are active. The leading edge initiates the start of the integration time and the falling edge defines the end of the integration time. The falling edge also initiates the readout period.

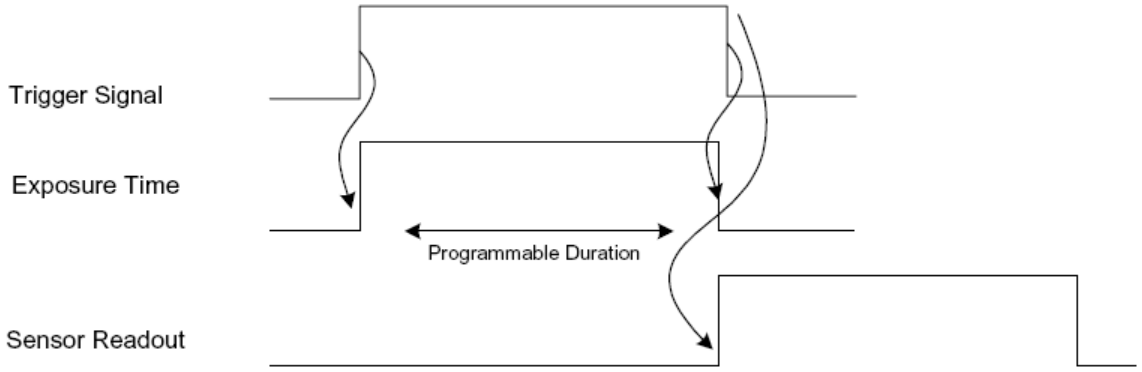


Figure 13.6 Integrate and Dump - Level Controlled Trigger Mode

13.4.2 Mode 1 Level Triggering for Interline Devices

In Level mode triggering the camera responds to a trigger edge by clearing the sensor photosites and letting the sensor integrate for a preset length of time. At the end of the integration time the image data is read from the sensor. The minimum time between triggers is approximately equal to the integration time plus the readout time.

13.4.3 Mode 1 Timing Sequence for Interline Devices

While waiting for a trigger, the sensor is continually being flushed of charge. When a trigger arrives via the rear panel the photosites are cleared and a strobe is issued out the rear panel. The sensor integrates for the duration of the trigger signal. During the integration period the sensor continues to be flushed of charge. When the integration time expires the image is read from the sensor.

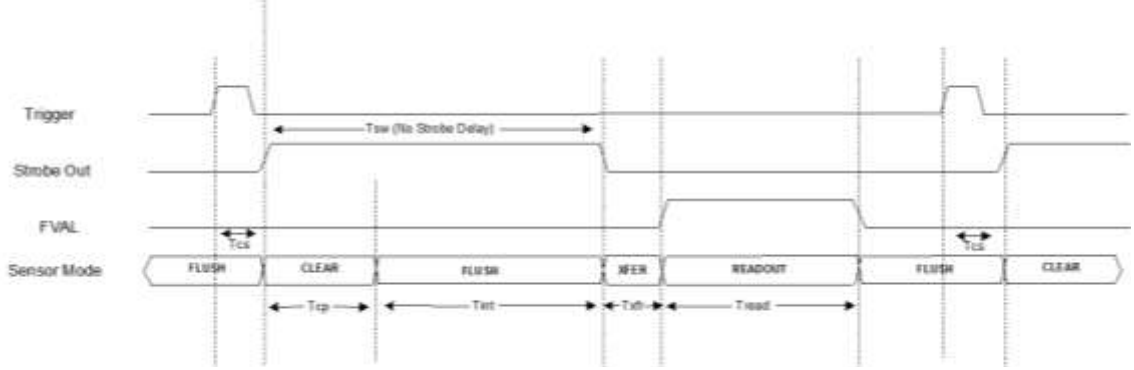


Figure 13.7 Triggering with No Strobe Delay

13.4.4 Mode 1 Timing Parameters for Interline Devices

Tcs (Clear Start Latency): This 2 µs interval defines the delay from the receipt of trigger to the start of the clear pulse to the sensor. This is the reaction time of the optoisolator.

Tsd (Strobe Delay): This interval defines and optional, user-adjustable delay from the start of the clear pulse to the issue of the strobe pulse. By default the strobe pulse begins at the time the clear pulse ends (i.e. Tsd = 0). The delay is under software control and is programmable from 0 to 1 second in 100 ns increments.

Tsw (Strobe Width): The duration of the strobe width equals the time of the clear pulse (Tcp) pulse the integration time (Tint) minus any optional strobe delay (Tsd). By default the strobe pulse begins at the time the clear pulse is sent to the sensor. The start of the strobe pulse may be delayed by a user-adjustable setting in the software.

Tcp (Clear Pulse): This is the time it takes to clear the sensor for interline sensors. The start of clear pulse begins as soon as possible after the camera receives a trigger (after Tcs). This time is sensor dependent. There is a nominal uncertainty in the start latency of +/- 1 pixel period.

Sensor	Sensor Frequency	Tcp
KAI-16000	30 MHz	15 µs
KAI-11002	30 MHz	19 µs
KAI-4020	30 MHz	19 µs
KAI-4020	38 MHz	15 µs
KAI-2093	30 MHz	25.3 µs
KAI-2093	38 MHz	20 µs
KAI-2001	30 MHz	25.3 µs
KAI-2001	38 MHz	20 µs

Table 13.7 Sensor/Sensor Frequency/Tcp

Tint (Integration Time): This is a user programmable interval which can range from 10 µs to 10 seconds. For interline sensors the absolute accuracy of the integration time is approximately 1 µs with an uncertainty of +/-1 pixel period.

Txfr (Transfer Time): This interval represents the time it takes to transfer the data from the photosites to the vertical shift register. This time is sensor dependent.

Sensor	Sensor Frequency	Txfr
KAI-16000	30 MHz	600 µs
KAI-11002	30 MHz	125 µs
KAI-4020	30 MHz	114 µs
KAI-4020	38 MHz	90 µs
KAI-2093	30 MHz	95 µs
KAI-2093	38 MHz	75 µs
KAI-2001	30 MHz	95 µs
KAI-2001	38 MHz	75 µs

Table 13.8 Sensor/Sensor Frequency/Txfr

Tread (Readout Time): This interval represents the time it takes to read the image from the sensor. This time is sensor dependent. The following readout times are measured results.

Sensor	Taps	Sensor Frequency	Tread
KAI-16000	1	30 MHz	588 ms
	2	30 MHz	322 ms
	1	38 MHz	476 ms
	2	38 MHz	256 ms
KAI-11002	1	30 MHz	392 ms
	2	30 MHz	212 ms
	1	38 MHz	309.5 ms
	2	38 MHz	167.3 ms
KAI-4020	1	30 MHz	153.7 ms
	2	30 MHz	82.8 ms
	1	38 MHz	121.4 ms
	2	38 MHz	65.4 ms
KAI-2093	1	30 MHz	76.5 ms
	2	30 MHz	41.4 ms
	1	38 MHz	60.4 ms
	2	38 MHz	32.7 ms
KAI-2001	1	30 MHz	72.7 ms
	2	30 MHz	40.6 ms
	1	38 MHz	57.4 ms
	2	38 MHz	32.1 ms

Table 13.9 Sensor/Sensor Frequency/ Single Tap/Dual Tap/Tread

Tr (Recovery Time): This interval represents the time from the end of sensor readout to the time the system can accept another trigger pulse. The recovery time is approximately 1 µs.

Maximum Trigger Rate: The maximum trigger rate is related to the minimum time between triggers which may be calculated as the sum of Ts + Tint + Txfr + Tread + Tr.

13.4.5 Mode 1 Level Triggering for Full Frame Sensors

In Level mode triggering, the camera responds to a trigger edge by stopping to flush the photosites and letting the sensor integrate for the duration of the trigger signal. At the end of the integration time the image data is read from the sensor. The minimum time between triggers is approximately equal to the integration time plus the readout time.

13.4.6 Mode 1 Level Triggered Timing Sequences for Full Frame Sensors

The following diagram illustrates the timing sequence for full frame sensors in Level triggered mode. While waiting for a trigger, the sensor is continually being flushed of charge. When a trigger arrives via the rear panel, the flushing stops and a strobe is issued out the rear panel. The sensor integrates for the duration of the trigger pulse. During the integration period the sensor clocking is halted. When the integration time expires the image is read from the sensor.

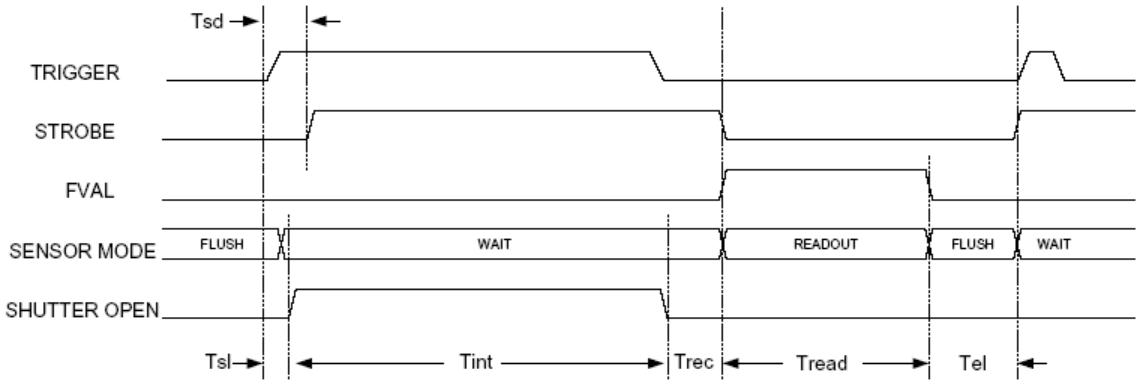


Figure 13.8 Level Mode Triggering for Full Frame Sensors with a Strobe Delay

13.4.7 Mode 1 Timing Parameters for Full Frame Sensors

The following definitions pertain to Figure 13.8.

Tcs (Clear Start Latency): This 2 µs interval defines the delay from the receipt of trigger to the start of the clear pulse to the sensor. This is the reaction time of the optoisolator.

Tsd (Strobe Delay): This interval defines the delay from the receipt of trigger to the issue of the strobe pulse. The delay is under software control and is programmable from 0 to 1 second in 100 ns increments.

Tsw (Strobe Width): The strobe pulse width is nominally 100 µs. The strobe polarity is under software control.

Ts (Start Latency): This is the time from the active edge of the trigger pulse to the start of the integration period. For full frame sensors the start latency is nominally equal to 10 pixel clocks.

Tint (Integration Time): The integration time is defined as the width of the trigger pulse minus the start latency. The accuracy of the integration time is controlled by the mechanical response of the shutter.

Tread (Readout Time): This interval represents the time it takes to read the image from the sensor. This time is sensor dependent.

Tr (Recovery Time): This interval represents the time from the end of sensor readout to the time the system can accept another trigger pulse. The recovery time is approximately 1 µs.

Maximum Trigger Rate: The maximum trigger rate is related to the minimum time between triggers which may be calculated as the sum of Trigger Width + Tread + Tr.

13.5 Mode 4; Double Exposure Triggering

13.5.1 Overview

The double exposure sequence begins after receiving a trigger input through the CameraLink connector or the TRIGGER connector on the rear panel of the camera. The camera captures the first image in the CCD and then transfers this image to the interline storage registers. The first image is transferred from the CCD to the internal frame store while the second image is being captured by the CCD.

Double exposure mode is a trigger mode in which the sensor rapidly acquires two images in response to a trigger and subsequently reads them out in sequence. The first image is acquired and transferred into the interline registers in response to the trigger. The second image is stored in the photosites and must wait for the first image to be read out before it can be transferred and read out. The sensor must operate in a dark environment with flash illumination, since after integrating the second image the data must wait in the photosite for the first image to be read out.

13.5.2 Double Exposure Mode Timing Sequence

Figure 13.9 illustrates the timing sequence for double exposure mode. While waiting for a trigger the sensor is continually being flushed of charge. When a trigger arrives via the rear panel the photosites are cleared and a strobe is issued out the rear panel. This signals the external electronics to flash the first round of illumination. The end of the Strobe Pulse occurs precisely at the TT end. There is a user programmable delay, TPD (transfer pulse delay), that allows time for the flash before the image is transferred into the interline registers. The TPD is approximately equal to the TT for most sensors and clock rates. *MEGAPLUS* Central will only allow input of a TDP that is less than the TT (Transfer Time). The falling edge of the Strobe will always reflect the time at which the first exposure ends and the second exposure begins. The IT (interframe time) is fixed at the smallest value based on the Camera Head sensor.

After the first image is transferred to the interline registers, readout begins. The external electronics (based on the values of TPD and IT) will flash the second round of illumination. This image is held in the photosites until the first image is read out. At this point the second image is transferred into the interline registers and read out.

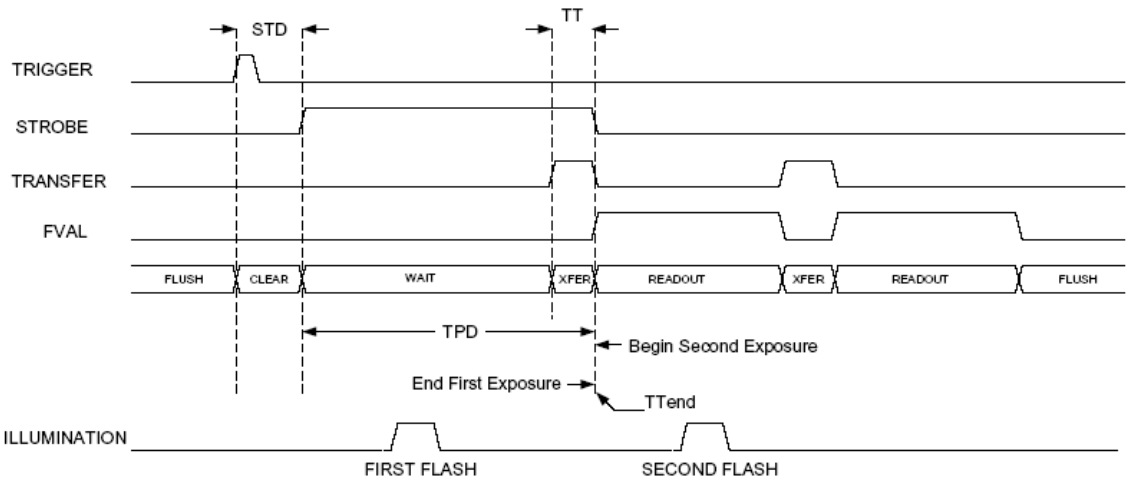


Figure 13.9 Double Exposure Timing

13.5.3 Double Exposure Mode Timing Parameters

The following definitions pertain to Figure 13.9.

STD (Strobe Delay): The strobe delay represents the time it takes for the camera to respond to the external trigger and issue a CLEAR pulse to the sensor.

Sensor	Pixel Clock Rate	STD
KAI-16000	30 MHz	22 µs
KAI-11002	30 MHz	17 µs
KAI-4020	30 MHz	21 µs
KAI-4020	38 MHz	17 µs
KAI-2093	30 MHz	27 µs
KAI-2093	38 MHz	22 µs
KAI-2001	30 MHz	26 µs
KAI-2001	38 MHz	21 µs

Table 13.10 Strobe Delay for Double Exposure Mode

TPD (Transfer Pulse Delay): a user programmable interval. Typical range is 1-1000,000 counts of 0.1 µs intervals. TPD value is a float value from 1 to 10,000 microseconds (10 µs) in 0.1 µs steps.

TT (Transfer Time): a fixed amount of time. The end of the transfer time is when the first exposure ends and the second exposure begins.

Sensor	Pixel Clock Rate	TT
KAI-16000	30 MHz	200 µs
KAI-11002	30 MHz	171 µs
KAI-4020	30 MHz	90 µs
KAI-4020	38 MHz	70 µs
KAI-2093	30 MHz	71 µs
KAI-2093	38 MHz	56 µs
KAI-2001	30 MHz	67 µs
KAI-2001	38 MHz	53 µs

Table 13.11 Transfer Time for Double Exposure Mode

Interframe Time: An interval that is fixed at the smallest value allowed by the Camera Head sensor. Each Camera Head will differ based on the sensor with a range starting from the low hundreds of nanoseconds to less than 1 microsecond.

First Image Integration Time: The integration time for the first image is equal to TPD.

Second Image Integration Time: The integration time for the second image is equal to the sensor readout time.

Maximum Trigger Rate: The maximum trigger rate is determined almost exclusively by the readout time of the sensor. Two readouts are required before the system can be triggered again.

Sensor	Pixel Clock	Taps	Frame Rate	Max Trigger Rate
KAI-16000	30 MHz	1	~1.7 Hz	~0.85 Hz
KAI-16000	30 MHz	2	~ 3.1 Hz	~1.55 Hz
KAI-11002	30 MHz	1	~2.5 Hz	~1.2 Hz
KAI-11002	30 MHz	2	~ 4.6 Hz	~2.3 Hz
KAI-4020	30 MHz	1	~ 6.3 Hz	~3.1 Hz
KAI-4020	30 MHz	2	~ 11.8 Hz	~ 6 Hz
KAI-4020	38 MHz	1	~ 8.1 Hz	~ 4 Hz
KAI-4020	38 MHz	2	~ 15 Hz	~7.5 Hz
KAI-2001	30 MHz	1	~13.6 Hz	~ 6.8 Hz
KAI-2001	30 MHz	2	~24.4 Hz	~ 12.2 Hz
KAI-2001	38 MHz	1	~17.2 Hz	~ 8.6 Hz
KAI-2001	38 MHz	2	~30.9 Hz	~15.5 Hz
KAI-2093	30 MHz	1	~12.9 Hz	~ 6.5 Hz
KAI-2093	30 MHz	2	~ 23.8 Hz	~ 11.9 Hz
KAI-2093	38 MHz	1	~ 16.3 Hz	~ 8.1 Hz
KAI-2093	38 MHz	2	~ 30.2 Hz	~15.1 Hz

Table 13.12 Maximum Trigger Rate for Double Exposure

13.6 Mode 6; Periodic Interval Triggering (Self Trigger)

13.6.1 Overview

When the camera is in trigger periodic interval mode, the camera self-triggers on a repeated cycle as long as the trigger mode is enabled. The interval is a user-defined value. In periodic pulse mode, triggering the camera responds to a software trigger edge by clearing the sensor photosites and letting the sensor integrate for a preset length of time. At the end of the integration time the image data is read from the sensor. The maximum pulse rate is related to the minimum time between triggers which is approximately equal to the integration time plus the readout time.

13.6.2 Periodic Pulse Timing Sequence

Figure 13.10 illustrates the timing sequence for interline sensors in periodic pulse trigger mode. While waiting for a trigger the sensor is continually being flushed of charge. When a trigger arrives via the rear panel, the photosites are cleared and a strobe is issued out of the rear panel. At the end of the CLEAR cycle a timer is started which has be pre-programmed with the desired integration time. During the integration period, the sensor continues to be flushed of charge. When the integration time expires the image is read from the sensor.

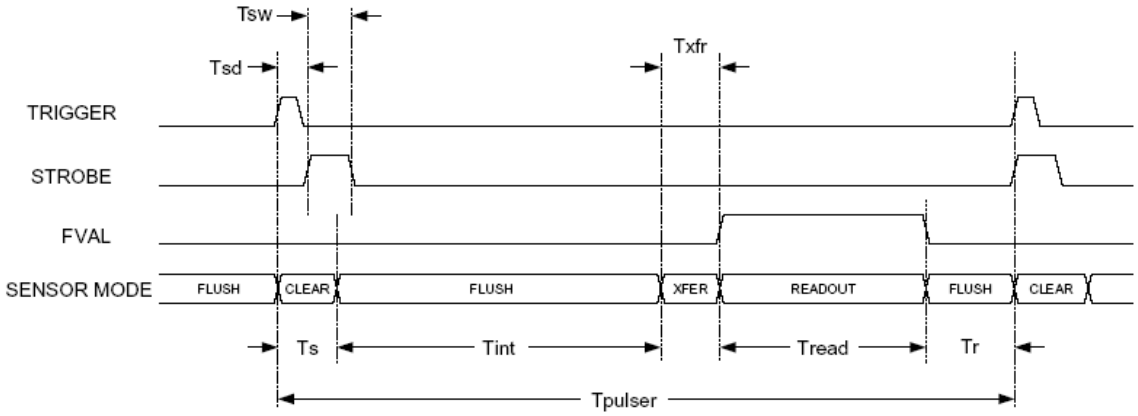


Figure 13.10 Periodic Interval Trigger Timing

13.6.3 Periodic Interval Timing Parameters

The following definitions pertain to Figure 13.10.

Tsd (Strobe Delay): This interval defines the delay from the receipt of trigger to the issue of the strobe pulse. The delay is under software control and is programmable from 1 µs to 1 second in 100 ns increments.

Tsw (Strobe Width): The strobe pulse width is nominally 100 µs. The strobe polarity is under software control.

Ts (Start Latency): This is the time from the leading edge of the trigger pulse to the start of the integration period. For interline sensors the start latency is equal to the time it takes to clear the sensor. This time is sensor dependent. There is a nominal uncertainty in the start latency of +/- 1 pixel period.

Sensor	Pixel Clock Rate	Ts
KAI-16000	30 MHz	15 µs
KAI-11002	30 MHz	19 µs
KAI-4020	30 MHz	19 µs
KAI-4020	38 MHz	15 µs
KAI-2093	30 MHz	25.3 µs
KAI-2093	38 MHz	20 µs
KAI-2001	30 MHz	25.3 µs
KAI-2001	38 MHz	20 µs

Table 13.13 Sensor/Pixel Clock Rate/Ts

Tint (Integration Time): This is a user programmable interval which can range from 10 µs to 10 seconds. For interline sensors the absolute accuracy of the integration time is approximately 1 µs with an uncertainty of +/- 1 pixel period.

Txfr (Transfer Time): This interval represents the time it takes to transfer the data from the photosites to the vertical shift register. This time is sensor dependent.

Sensor	Pixel Clock Rate	Txfr
KAI-16000	30 MHz	600 µs
KAI-11002	30 MHz	125 µs
KAI-4020	30 MHz	114 µs
KAI-4020	38 MHz	90 µs
KAI-2093	30 MHz	95 µs
KAI-2093	38 MHz	75 µs
KAI-2001	30 MHz	95 µs
KAI-2001	38 MHz	75 µs

Table 13.14 Sensor/Pixel Clock Rate/Txfr

Tread (Readout Time): This interval represents the time it takes to read the image from the sensor. This time is sensor dependent.

Sensor	Taps	Sensor Frequency	Tread
KAI-16000	1	30 MHz	588 ms
	2	30 MHz	322 ms
	1	38 MHz*	476 ms
	2	38 MHz*	256 ms
KAI-11002	1	30 MHz	392 ms
	2	30 MHz	212 ms
	1	38 MHz*	309.5 ms
	2	38 MHz*	167.3 ms
KAI-4020	1	30 MHz	153.7 ms
	2	30 MHz	82.8 ms
	1	38 MHz	121.4 ms
	2	38 MHz	65.4 ms
KAI-2093	1	30 MHz	76.5 ms
	2	30 MHz	41.4 ms
	1	38 MHz	60.4 ms
	2	38 MHz	32.7 ms
KAI-2001	1	30 MHz	72.7 ms
	2	30 MHz	40.6 ms
	1	38 MHz	57.4 ms
	2	38 MHz	32.1 ms

* Readout time in Overlap triggering mode will be longer.

Table 13.15 Sensor/Taps/Sensor Frequency/Tread

Tr (Recovery Time): This interval represents the time from the end of sensor readout to the time the system can accept another trigger pulse. The recovery time is approximately 1 µs.

Tpulser: This is the interval between software triggers and is user programmable from 1 ms to 10 seconds in 1 ms increments. The absolute accuracy of this interval is +/-1 ms.

Maximum Pulser Rate: The maximum pulser rate is related to the minimum time between triggers which may be calculated as the sum of Ts + Tint + Txfr + Tread + Tr.

13.7 Mode 7; Overlap Triggering (Asynchronous Reset with Programmable Integration and Readout During Integration)

13.7.1 Overview

The Overlap Triggering Mode function is similar to Mode 0. The exception is, when an incoming trigger arrives during the image readout, the camera starts a new frame without interrupting the current image readout. The camera responds to a trigger edge by clearing the sensor photosites and letting the sensor integrate for a preset length of time. At the end of the integration time the image data is read from the sensor. With Overlap Triggering a new trigger signal is allowed to start a new integration period while the previous frame is being read out. The time between triggers and the programmed integration time must be set so that the readout of the previous frame concludes by the end of the integration period for the first frame.

NOTE: A firmware update may be required for proper functionality of overlap triggering.

13.7.2 Edge Triggered Overlapped Timing Sequence

Figure 13.11 illustrates the timing sequence for sensors in Overlap Trigger mode. While waiting for a trigger the sensor is continually being flushed of charge. When a trigger arrives via the rear panel the photosites are cleared and a strobe is issued out the rear panel. At the end of the CLEAR cycle, a timer is started which has be pre-programmed with the desired integration time. During the integration period the sensor continues to be flushed of charge. When the integration time expires, the image is read from the sensor. If another trigger pulse arrives during readout of the sensor, a CLEAR cycle is inserted into the next line of the READOUT cycle. The integration timer is then restarted. When the readout of the current frame expires, the sensor enters FLUSH mode again and waits for the integration timer to expire.

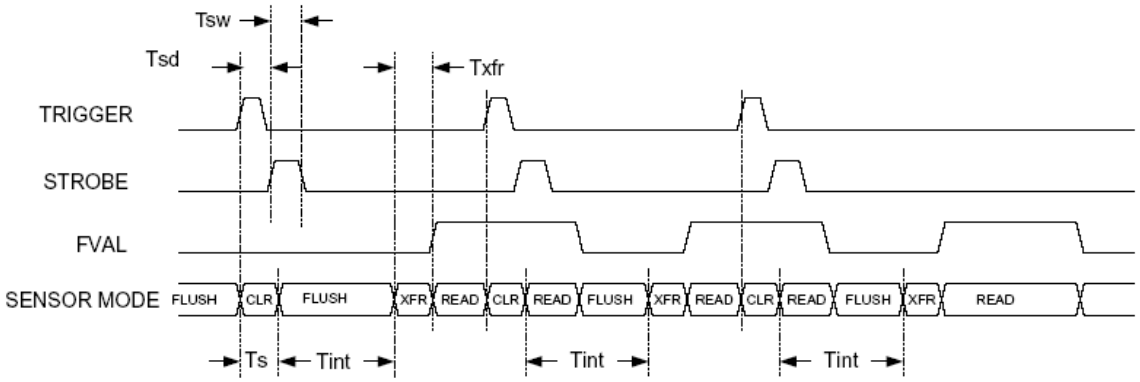


Figure 13.11 Overlap Trigger Mode Timing

13.7.3 Edge Triggered Overlapped Timing Parameters

The following definitions pertain to Figure 13.11.

Tsd (Strobe Delay): This interval defines the delay from the receipt of trigger to the issue of the strobe pulse. The delay is under software control and is programmable from 1 µs to 1 second in 100 ns increments.

Tsw (Strobe Width): The strobe pulse width is nominally 100 µs. The strobe polarity is under software control.

Ts (Start Latency): This is the time from the leading edge of the trigger pulse to the start of the integration period. For interline sensors the start latency is equal to the time it takes to clear the sensor. This time is sensor dependent. There is a nominal uncertainty in the start latency of +/- 1 pixel period. For trigger signals which arrive during the readout of the previous frame, there is an additional start latency which can vary from 0 to one line length in duration. This is because the CLEAR cycle which results from the trigger input must wait for the current line to be read out before it can be executed. The line duration is sensor dependent as well as clock rate and single/dual tap mode dependent.

Sensor	Pixel Clock Rate	Ts
KAI-16000	30 MHz	15 µs
KAI-11002	30 MHz	19 µs
KAI-4020	30 MHz	19 µs
KAI-4020	38 MHz	15 µs
KAI-2093	30 MHz	25.3 µs
KAI-2093	38 MHz	20 µs
KAI-2001	30 MHz	25.3 µs
KAI-2001	38 MHz	20 µs

Table 13.16 Sensor/Pixel Clock Rate/Ts

Tint (Integration Time): This is a user programmable interval which can range from 10 µs to 10 seconds. For interline sensors the absolute accuracy of the integration time is approximately 1 µs with an uncertainty of +/- 1 pixel period.

Txfr (Transfer Time): This interval represents the time it takes to transfer the data from the photosites to the vertical shift register. This time is sensor dependent.

Sensor	Pixel Clock Rate	Txfr
KAI-16000	30 MHz	600 µs
KAI-11002	30 MHz	125 µs
KAI-4020	30 MHz	114 µs
KAI-4020	38 MHz	90 µs
KAI-2093	30 MHz	95 µs
KAI-2093	38 MHz	75 µs
KAI-2001	30 MHz	95 µs
KAI-2001	38 MHz	75 µs

Table 13.17 Sensor/Pixel Clock Rate/Txfr

Tread (Readout Time): This interval represents the time it takes to read the image from the sensor. This time is sensor dependent.

Sensor	Taps	Sensor Frequency	Tread
KAI-16000	1	30 MHz	588 ms
	2	30 MHz	322 ms
	1	38 MHz*	476 ms
	2	38 MHz*	256 ms
KAI-11002	1	30 MHz	392 ms
	2	30 MHz	212 ms
	1	38 MHz*	309.5 ms
	2	38 MHz*	167.3 ms
KAI-4020	1	30 MHz	153.7 ms
	2	30 MHz	82.8 ms
	1	38 MHz	121.4 ms
	2	38 MHz	65.4 ms
KAI-2093	1	30 MHz	76.5 ms
	2	30 MHz	41.4 ms
	1	38 MHz	60.4 ms
	2	38 MHz	32.7 ms
KAI-2001	1	30 MHz	72.7 ms
	2	30 MHz	40.6 ms
	1	38 MHz	57.4 ms
	2	38 MHz	32.1 ms

* Running the KAI-16000 or KAI-11002 in an over clocked state (38 MHz) is not recommended.

Table 13.18 Sensor/Taps/Sensor Frequency/Tread

Maximum Trigger Rate: The maximum trigger rate is related to the minimum time between triggers. In Overlap mode the only constraint on the time between triggers is that the readout of the previous frame must have expired before the readout of the current frame starts.

13.8 Retriggering

13.8.1 Overview

Should a second trigger arrive before the processing of the first trigger is complete, there are two possible responses by the camera. The camera's response depends on when in the triggering process the second trigger arrives. Retriggering is not available in Overlap Triggering mode. The maximum trigger rate constraints must not be violated.

13.8.2 Retriggering Timing Sequence

Figure 13.12 illustrates the camera's response to the second trigger if it arrives during the integration cycle of the first trigger. In this case, the camera resets and the triggering process starts over.

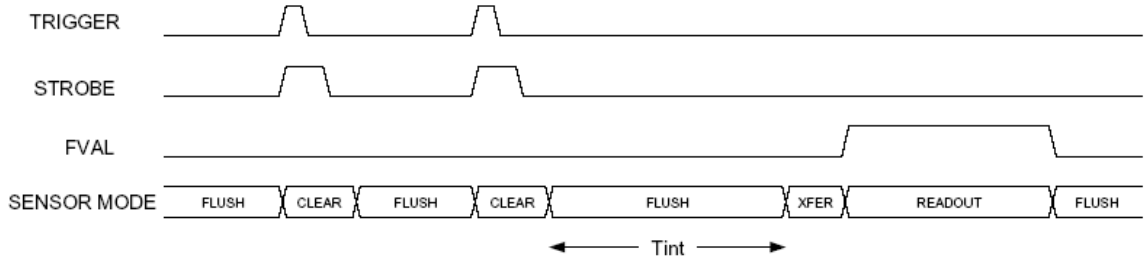


Figure 13.12 Retriggering of a Camera during Integration

Figure 13.12 illustrates the camera's response to a second trigger if it arrives during the transfer or readout cycle of the first trigger. In this case, the second trigger is "latched" and processed after the current trigger cycle finishes.

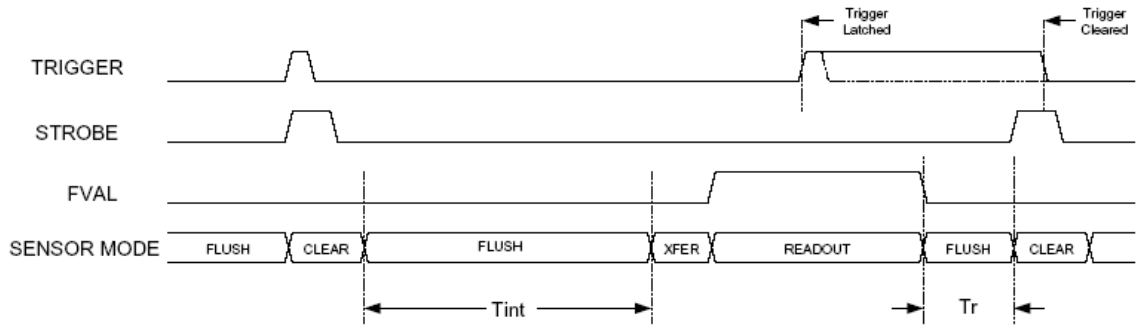


Figure 13.13 Retriggering of a Camera during Transfer or Readout

Tint (Integration Time): This is a user programmable interval which can range from 10 µs to 10 seconds. For interline sensors the absolute accuracy of the integration time is approximately 1 µs with an uncertainty of +/- 1 pixel period.

Tr (Recovery Time): This interval represents the time from the end of sensor readout to the time the system can accept another trigger pulse. The recovery time is approximately 1 µs.

14. Serial Command Protocol

14.1 Introduction

The *MEGAPLUS* camera platform supports user control of camera configuration and operation via an RS-232 serial communications link. The camera allows serial input via one of two serial ports: a) the 9-pin D connector on the rear panel of the Camera Controller or b) the serial communication link that is embedded in the CameraLink data interface.

When the rear panel serial connection is utilized, the port can be controlled from the host PC via one of the standard system serial ports (for example COM1 and COM2). When the CameraLink serial interface is used, programmatic access to port I/O will depend on the tools your frame grabber vendor provides. Some frame grabber vendors provide a utility that maps the CameraLink serial port to look like one of the system serial ports. In this case, once the CameraLink serial port has been mapped, it can be accessed programmatically as COM1, COM2, etc. If your frame grabber vendor supports the AIA CameraLink serial port Application Programming Interface (API), the CameraLink serial port can be accessed through the CameraLink Dynamic Link Library.

Regardless of the serial port option you select, the serial port communications must be configured to the follow conditions:

Serial Port Configuration

- Baud Rate: 9600
- Parity: None
- Data Bits: 8
- Stop Bits: 1
- XON/XOFF Mode: Disabled
- CTS Hardware Handshaking: Disabled

Once the serial communication link is open, camera control is implemented with an ASCII control protocol in which command strings are sent to the camera and it responds with a command echo, return values, and status information. The message formats (command and response) for each command are detailed below.

14.2 Command Syntax

In the nomenclature below <LF> indicates the "carriage return" termination character designated as the terminating value for the command string.

IMPORTANT: Each part of a command, the command number and any parameter values must be separated by a space (" ") character. This white space character delimits each part of the command.

For each command, the camera will return a status value in response to a command. Possible status values and their meanings are as follows:

0 = OK, command executed without problem

-1 = Error, some failure occurred

-2 = Argument out of range

-3 = Feature not supported. The requested feature is not supported in this configuration of the camera.

-4 = Warning: the command executed, but there may be some effects on camera performance.

Camera commands generally fall into one of a few categories. A given command may request configuration information, control an attribute of a specific Camera Head, or control a general feature of the camera controller that does not affect Camera Head operation, or applies the same to all Camera Heads.

When specifying commands that apply to a Camera Head rather than the Camera Controller, is it necessary to define which Camera Head the command is to be applied to. This can occur implicitly or explicitly.

In any Camera Head related control function, the target head explicitly specified is the last parameter in the parameter list. This is the safest, most definite manner to specify which Camera Head is the recipient of the command. The head number specified is a 1-based index between 1 and the maximum number of Camera Heads the controller supports (typically 4).

If no head number is included, the function is applied to the currently active head.

The target Camera Head can be defined for a long series of commands by using the SetActiveHead() function. Once a given head has been specified as the active head, all head-related control functions issued without an explicit head number in the parameter list will be targeted for that "active" Camera Head.

NOTE: In cameras with only one Camera Head that head will always be the currently active head.

Some commands may target a sub-class of the Camera Head object. For example, some Camera Heads may support more than one sensor. Likewise, some sensors may have more than one tap readout. For functions that apply to a specific sensor, or a single tap of a sensor, the command target object will always be explicitly specified in the parameter list for the function. The target object specification will consist of the following: <head no>, <sensor no>, <tap no>.

The majority of the sensors supported by this platform will have only one or two taps. In a two-tap sensor, the taps are generally referred to as "left" or "right" taps. This is because the taps are configured to read out the left half or right half of the image. However, some Camera Heads may utilize sensors that have more than two taps for example, linescan sensors or high-speed sensors. In order to support these sensor types, the tap specification will be as follows: 0 = all taps (for example both taps of a 2-tap sensor or all taps of a multi-tap sensor). For individual taps, a numerical value specifies the tap number of a sensor for example, 8 would be the 8th tap of a multi-tap sensor. In a 2-tap sensor, tap 1 is the "left" tap and tap 2 it the "right" tap. If a value larger than the maximum tap number is passed, the value will be ignored and a value of 0 will be applied.

14.3 Camera Response to Commands

NOTE: In the ASCII strings received from the camera, each value is separated with an ASCII "space" character.

Whenever the camera receives a command it will respond with the following protocol:

<command echo> <status> <no retvals> <retval1> … <retval n> <LF>

The return message will echo the command number that it is responding to <command echo> followed by a status value <status>.

If the status is OK, the next parameter specifies the number of return data values to follow. This parameter count is followed by each of the return values and finally the <LF> terminator character. If the command failed, the response will be simply:

<command echo> <status><LF>.

14.4 Nomenclature

In the documentation that follows:

<param>: indicates a parameter that is passed with a command. The actual parameter value is passed in ASCII format. For example the value 9.3 would be represented in the command as the ASCII characters "9.3".

(Param): Indicates an optional parameter in the command line.

14.5 Configuration Functions

These commands are used to determine or control configuration attributes of the camera.

`GetVersion()`: Queries camera to return current firmware version information.
Command String: "101<LF>"
Response: "101 <status> 4<ret_cnt> <fw_ver> <fpga_ver> <serial_prot_ver><LF>"
Where: 4 => return parameter count
<ret_cnt> => count of the number of values to follow
<fw_ver> => camera firmware version in the format nn.mm
<fpga_ver> => FPGA version in the format nn.mm
<serial_prot_ver> => Serial protocol version in the format nn.mm

The version number format is nn.mm where nn is the major version number, and mm is the minor version number.

`GetMaxHeadsSupported()`: Returns the maximum number of heads that this controller can support.
Command String: "102<LF>"
Response: "102 <status> 1 <n><LF>""
Where: 1 => return parameter count
<n> => max heads supported (range 1-4)
2 = Dual Head Color FPGA Configuration
4 = for Four Head Mono FPGA Configuration

`GetNoHeadsAttached()`: Queries the camera and returns the number of heads currently attached to the controller.
Command String: "103<LF>"
Response: "103 <status> 1 <n><LF>"
Where: 1 => return parameter count
<n> => number of heads currently attached (range 1-4)

`GetHeadCfg(headno)`: Returns the current configuration information for the specified Camera Head.
Command String: "104 <hdno><LF>"
Where: hdno = 1-4
Response: "104 <status> 6 <no_sensors> <no_tapsavail> <spec_type> <h_size> <v_size> <ser_no><LF>"
Where:
No_Sensors: (0 => None-no head present,
1 => 1-CCD head present
2 => 2-CCD head present (possible in the future)
3 => 3-CCD head present)
No_TapsAvail per sensor: (1 or 2 now, more in the future)
Spec Type: (Mono, RGB, or CIR)
H_Size: Horizontal pixels
V_Size: Vertical pixels
Ser_No : Head serial number

`SetActiveHead(headno)`: (Unused for the single-head controller.) Specifies the head to be designated as the currently active head. If this command is successful, all subsequent commands that apply to camera heads will affect the specified head. If the command fails, the currently active head remains unchanged from its previous setting.

NOTE: This command does NOT control what image data is output from the camera. To select output of the images from a specific head use the QuickMux command. For CameraLink interfaces, the CLMux command can also be used to customize output.

Command String: "105 <hdno><LF>"
Where: hdno => 1-4
Response: "105 <status> 0<LF>"
Where: 0 => return parameter count

`GetActiveHead()`:(Unused for the single-head controller.) Queries the camera for the head number of the currently active head. Returns zero if no heads are attached.
Command String: "106<LF>"
Response: "106 <status> 1 <hdno> <LF>"
Where: 1 => return parameter count
<hdno> => 1-4

`SetCfgSelect(cfg_no)`: (Unused for the single-head controller.) Sets a value in the camera that will determine which configuration the camera will boot as the next time the controller is powered on or off. The configuration is specified by an index number that is passed as a parameter. This selection flag is written to the camera's non-volatile memory. The next time the camera is turned on, this value will be read and used to control the way the camera configures itself.

NOTE: The MEGAPLUS camera can support a variety of different configurations and features depending on the firmware contents that are loaded (FPGA configuration and processor software). This flag is used to tell the camera which version of the firmware is loaded the next time the camera is started.

Command String: "107 <cfg_no><LF>"
Where: cfg_no => a configuration ID number.
1 = Dual-head Advanced Color
2 = Four-head Mono (Not with EP/EC 16000)
3 = Dual-head Mono
4 = Single-head Controller
Response: "107 <status> 0<LF>"
Where: 0 => return parameter count

`GetCfgSelect(curr_cfg, cfg_flag )`: (Unused for single-head controller.) Queries the camera to return the value of the currently loaded configuration index number and the value of the configuration selection flag that will determine the configuration the next time the camera power is cycled. If the camera has received a new configuration selection value via the SetCfgSelect() command, but the camera has not yet been powered down, these two values can be different. If the configuration selection has not been changed since the camera was last booted, these two values should be the same.
Command String: "108<LF>"
Response: "108 <status> 4 <curr_cfg> <cfg_flag><LF>"
Where: <curr_cfg> => the currently loaded configuration no.
1 = Dual-head Advanced Color
2 = Four-head Mono (Not with EP/EC 16000)
3 = Dual-head Mono
4 = Single-head Controller
<cfg_flag> => the value of the configuration flag that determines the configuration loaded on next boot cycle

14.6 Camera Head Related Functions

In the following functions, the target Camera Head can be specified explicitly in the command string or implicitly with the SetActiveHead command.

14.6.1 Gain

`Properties_Gain()`: Returns the properties of the gain feature for the selected Head.
Properties include: IsSupported, min, max, one shot avail, auto avail.
Command String: "201 (hd_no)<LF>"
Response: "201 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max gain value in dB (x.xx format), (32.00 dB)
<min> => min gain value in dB (x.xx format), (0.00 dBl)
<one_shot> 1=one shot supported, 0=not supported (0)
<auto> => 1=auto mode supported, 0=not supported (0)

`SetGain(gainval)`: Sets the head gain for the selected Head to the specified gain value in dB. If hd_no is not specified, sets the gain for the currently active head.
Command String: "202 <value> (hd_no)<LF>"
Where: <value> is the floating point gain value in dB (x.xx format). Must fall within min and max specified in Properties_Gain command.
Response: "202 <status> 0<LF>"
Where: 0 => return parameter count

`GetGain(gainval)`: Returns the head gain for the selected Head in dB. If hd_no is not specified, sets the gain for the currently active head.
Command String: "203 (hd_no)<LF>"
Response: "203 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> = gain value in dB returned from the camera.

14.6.2 Brightness (Offset)

`Properties_Brightness()`: Returns the properties of the brightness feature for the selected Head. Brightness is also referred to as "Offset". It is a constant digital number that adds an offset to each pixel in the image. Properties include: IsSupported, min, max, one shot avail, auto avail. If hd_no is not specified, returns the properties for the currently active head.
Command String: "206 (hd_no)<LF>"
Response: "206 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1= is supported, 0= not supported
<max> => max brightness value in DN, (255)
<min> => min brightness value in DN, (0)
<one_shot> 1=one shot supported, 0=not supported
<auto> => 1=auto mode supported, 0=not supported

`SetBrightness(countval)`: Sets the brightness or offset for the selected Head to the specified value in DN counts. If hd_no is not specified, sets the brightness for the currently active head.
Command String: "207 <value> (hd_no)<LF>"
Where: <value> is the brightness value in DN. Must fall within min and max specified in Properties_Brightness() command.
Response: "207 <status> 0<LF>"
Where: 0 => return parameter count

`GetBrightness(countval)`: Returns the brightness for the selected Head in DN counts. If hd_no is not specified, returns the brightness for the currently active head.
Command String: "208 (hd_no)<LF>"
Response: "208 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> = brightness value in DN returned from the camera.

14.6.3 Integration Time

`Properties_IntTime()`: Returns the properties of the integration time feature for the selected Head. Integration time is the amount of time the sensor collects photons for a single acquisition. Properties include: IsSupported, min, max, one shot avail, auto avail. If hd_no is not specified, returns the properties for the currently active head.
Command String: "209 (hd_no)<LF>"
Response: "209 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1= is supported, 0= not supported
<max> => max shutter value in msec (x.xx format)
<min> => min shutter value in msec (x.xx format)
<one_shot> 1=one shot supported, 0=not supported
<auto> => 1=auto mode supported, 0=not supported

Important Things to Know

MEGAPLUS offers two integration time controls: one for free-run integration time and one for triggered operation. If you are running the camera in triggered mode, you must use the **`MP_SetTriggerIntTime()`** function to control integration time.

The range of integration time values available is determined by 1) tap readout configuration, 2) pixel clock speed, and binning level. Any time that you change any of these parameters, the maximum and minimum free-run integration time values will also change. Therefore, whenever you change readout (**`MP_SetSensorTapReadout()`**), speed (**`MP_SetCamPixClkSpeed()`**), or binning level (**`MP_SetBinning()`**) you should

use the **`MP_PropertiesIntTime()`** function to re-query the available integration time range.

`SetIntTime(int_time)`: Sets the free-run integration time for the selected Head to the specified value in msec. All heads use the last entered integration time unless "boot configuration" is set to "Dual-head Advanced Color" (see command 107). This command controls integration time for only free-run, continuous video, when "trigger state" is set to 0. See command 410 to control integration time while using the camera in a trigger mode.
Command String: "210 <value> (hd_no)<LF>"
Where: <value> is the floating point integration time in msec (x.x format). Must fall within min and max specified in Properties_Shutter command. If a value larger than the maximum is specified, the integration time will be set to the maximum allowable value.
Response: "210 <status> 0<LF>"
Where: 0 => return parameter count

`GetIntTime(int_time)`: Returns the free-run integration time for the selected Head in msec. All heads use the last entered integration time unless "boot configuration" is set to "Dual-head Advanced Color" (see command 107). This command controls integration time for only free-run, continuous video, when "trigger state" is set to 0. See command 410 to control integration time while using the camera in a trigger mode.
Command String: "211 (hd_no)<LF>"
Response: "211 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> = shutter time in msec returned from the camera.

14.6.4 White Balance

`Properties WhiteBalance`: Returns the properties of the "whitebalance" feature for the selected Head. Properties include: IsSupported, min, max, one shot avail, auto avail. Applies only to color sensors. If hd_no is not specified, returns the properties for the currently active head. Note: If the camera is white-balance capable for this camera head series, it will return a value of 1 for is_supported even if a monochrome camera head is attached.
Command String: "217 (hd_no)<LF>"
Response: "217 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max white bal value (64)
<min> => min white bal value (1)
<one_shot> 1=one shot supported, 0=not supported
<auto> => 1=auto mode supported, 0=not supported

`SetWhiteBalance(one_push, red_val, blue_val)`: Sets the white balance for the currently selected head. If the one_push argument is true, the input values of red_val and blue_val arguments are ignored and the camera will perform a semi-auto white balance. Upon return the red_val and blue_val arguments will contain the resulting white balance parameters. If the one_push argument is false (manual white balance,) the red_val and blue_val arguments specify the relative settings of the component signals. The values specified for red and blue are relative to green, which is the master color. The value for green is always 1.0. If hd_no is not specified, sets the white balance for the currently active head. **Note:** The whitebalance process will function, even when applied to a monochrome head, but will have no effect on the data. Semi-auto whitebalance on a monochrome head will return values of 1.00 for the red and green color values.

NOTE: The one-shot automatic white balance algorithm in the camera processor repeatedly acquires an image and adjusts color gain parameters until the best achievable balance is determined. The time required for this

NOTE: The *GetAvgValues()* function can be used to read the average pixel value within the white balance reference area in order to assess how well-balanced the camera is.

Command String: "218 <one_shot> <red_val> <blue_val> (hd_no)<LF>"
Where: <one_shot> = 0, manual balance, apply specified values and = 1, camera will automatically determine adjustment values
<red_val> = relative strength of the red signal relative to the green. Must fall within min and max specified in Properties_WhiteBalance command. Values may be formatted with an "m.n" decimal format, but will be rounded to nearest integral value.
<blue_val> = relative strength of the blue signal relative to green. Must fall within min and max specified in Properties_WhiteBalance command. Values may be formatted with an "m.n" decimal format, but will be rounded to nearest integral value.
Response: "218 <status> 2 <red_val> <blue_val> <LF>"
Where: 2 => return parameter count
<red_val> = relative strength of red signal
<blue_val> = relative strength of blue signal

`GetWhiteBalance(one_push, red_val, blue_val)`: Returns the current white balance settings for the selected Head. If hd_no is not specified, returns the whitebalance for the currently active head.
Command String: "219 (hd_no)<LF>"
Response: "219 <status> 2 <red_val> <blue_val><LF>"
Where: 2 => return parameter count
<red_val> = strength of red values relative to green in "m.n" format
<blue_val> = strength of blue values relative to green in "m.n" format

14.6.5 Camera Head Temperature Control

The serial commands in this section deal with determining the cooling method in the selected camera (whether the camera is passively cooled – passive airflow; or actively cooled – TEC or fan), setting and getting the temperature in the camera head, setting/getting a TEC target temperature, turning a TEC on/off, and turning a fan on/off.

`Properties_HeadTemp`: Returns the properties of the camera head temperature feature for the selected Head, use the *Properties_Head Sensor TEC Temp command*. If hd_no is not specified, returns the properties for the currently active head. Properties include: IsSupported, min, max, one shot avail, auto avail. The max and min values indicate the temperature sensor range and are not an indication of the camera head operating temperature range.
Command String: "220 (hd_no)<LF>"
Response: "220 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => N/A: value = Maximum value (250)
<min> => N/A: value = Minimum value (-100)
<one_shot> N/A: value = 0
<auto> => N/A: value = 0

`GetHeadTemp(temp)`: This command returns the temperature of the specified Camera Head. Used for cameras that do not contain a TEC. For cameras containing a TEC, refer to the TEC-related commands (for example, *GetHeadSensorTECTemp (temp),* command 251). If hd_no is not specified, returns the temperature for the currently active head.
Command String: "221 (hd_no)<LF>"
Response: "221 <status> 1 <temp><LF>"
Where: 1 => return parameter count
<temp> = temperature of the Camera Head in degrees Celsius. Reported in "m.n" format.

`Properties_Head Sensor TEC Temp`: Returns the properties of the camera head sensor TEC temperature feature for the selected Head. Use the SetHeadSensorTECState command to turn the TEC on/off or set it to Auto. Properties include: IsSupported, min, max, one shot avail, auto avail. The only properties applicable to this feature is the IsSupported, max value, and min value property. One-Shot and Auto are not applicable and return a value of 0.
Command String: "250 (hd_no)<LF>"
Response: "250 <status> 5 <is_sup><max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => N/A: value = 0
<min> => N/A: value = 0
<one_shot> N/A: value = 0
<auto> => N/A: value = 0

`GetHeadSensorTECTemp(temp)`: This command returns the temperature of the specified camera head sensor TEC.
Command String: "251 (hd_no)<LF>"
Response: "251 <status> 1 <temp><LF>"
Where: Status: 0 = OK, -1 = not available for this head (available only for EC heads),
1 => return parameter count
<temp> = temperature of the camera head in degrees Celsius

`Properties_HeadSensorTECTargetTemp`: Returns the properties of the camera head sensor TEC target temperature feature for the selected Head. Properties include: IsSupported, min, max, one shot, and auto. The only properties applicable to this feature are IsSupported, max value, and min value. One-Shot and Auto are not applicable and return a value of 0.
Command String: "252 (hd_no)<LF>"
Response: "252 <status> 5 <is_sup><max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => maximum value supported
<min> => minimum value supported
<one_shot> N/A: value = 0
<auto> => N/A: value = 0

`SetHeadSensorTECTargetTemp(temp)`: This command sets the target temperature of the specified camera head sensor TEC. This target temperature setting will be only used if SetHeadSensorTECState (command 256) has set the TEC state to Auto.
Command String: "253 <temp> (hd_no)<LF>"
Where: <temp> = target temperature of the camera head TEC in degrees Celsius
Response: "253 <status> 0 <LF>"
Where: <status> => 0 = OK, -1 = not available for this head, 0 => return parameter count

`GetHeadSensorTECTargetTemp`: This command returns the target temperature of the specified camera head sensor TEC. Use *GetHeadSensorTECTemp (command 251) to get the actual temperature.*
Command String: "254 (hd_no)<LF>"
Response: "254 <status> 1 <temp><LF>"
Where: <status> => 0 = OK, -1 = not available for this head,
1 => return parameter count
<temp> = target temperature of the camera head TEC in degrees Celsius

`Properties_HeadSensorTECState`: Returns the properties of the camera head sensor TEC On/Off state for the selected head. Properties include: IsSupported, min, max, one shot, auto. One shot and auto are not applicable.
Command String: "255 (hd_no)<LF>"
Response: "255 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => maximum value supported (1 or 2 if TEC is present, dependent on type of TEC, else 0)
<min> => minimum value supported (0)
<one_shot> => N/A, value = 0
<auto> => N/A, value = 0

`SetHeadSensorTECState (state)`: This command turns the TEC on or off for the specified camera head. The TEC state must be set to Auto if you want to activate a target temperature setting entered via the SetHeadSensorTECTargetTemp command (command 253).
Command String: "256 <state> (hd_no)<LF>"
Where: <state> => on/off state of the TEC (0 = off, 1 = on, 2 = auto)
Response: "256 <status> 0 <LF>"
Where: <status> : 0 = OK
0 => return parameter count

`GetHeadSensorTECState`: This command returns the current on or off state of the TEC for the specified camera head.
Command String: "257 (hd_no)<LF>"
Response: "257 <status> 1 <state><LF>"
Where: <status> : 0 = OK
1 => return parameter count
<state> on/off state of the camera head TEC returned from camera (0 = off, 1 = on, 2 = Auto)

`Properties_HeadSensorFanState`: Returns the properties of the camera head sensor fan On/Off state for the selected head. Properties include: IsSupported, min, max, one shot, auto. One shot and auto are not applicable.
Command String: "258 (hd_no)<LF>"
Response: "258 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => maximum value supported (1 if fan is present, else 0)
<min> => minimum value supported (0)
<one_shot> => N/A, value = 0
<auto> => N/A, value = 0

`SetHeadSensorFanState (state)`: This command turns the head cooler fan on/off state for the specified camera head.
Command String: "259 <state> (hd_no)<LF>"
Where: <state> => on/off state of the fan (0 = off, 1 = on)
Response: "259 <status> 0 <LF>"
Where: <status> : 0 = OK
0 => return parameter count

`GetHeadSensorFanState`: This command returns the current on or off state of the head cooler fan for the specified camera head.
Command String: "260 (hd_no)<LF>"
Response: "260 <status> 1 <state><LF>"
Where: <status> : 0 = OK
1 => return parameter count
<state> On/Off state of the camera head cooler fan returned from camera (0 = off, 1 = on)

14.6.6 Crosshairs

`Properties_CrossHairs()`: Returns the properties of the crosshairs feature for the selected Head. If hd_no is not specified, returns the properties for the currently active head. Properties include: IsSupported, min, max, one shot avail, auto avail. The one shot avail and auto avail modes are not applicable.
Command String: "222 (hd_no)<LF>"
Response: "222 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max value (1)
<min> => min value (0)
<one_shot> = 0=not supported
<auto> = 0=not supported

`SetCrossHairs(on_off)`: Turns the crosshair display for the currently selected head on or off. If hd_no is not specified, sets the crosshairs for the currently active head. The crosshair boundary indicates the region of the image used for semi-auto white balance calculations. The indicator is positioned in the exact digital center of the image and is useful for camera targeting.

WARNING: The crosshair display is produced by setting pixel values in the image to display the crosshairs, therefore it supersedes the original image pixel values. You should be sure to remember to disable the display of the crosshair when you are ready to acquire original images.

Command String: "223 <bool> (hd_no)<LF>"
Where: <bool> =>1 = crosshairs on, 0 = crosshairs off
Response: "223 <status> 0<LF>"
Where: 0 => return parameter count

`GetCrossHairs(on_off)`: Returns the current state of the crosshair display for the currently selected head.
Command String: "224 (hd_no)<LF>"
Response: "224 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> => 1 = crosshairs on, 0 = crosshairs off

14.6.7 Mechanical Shutter

`Properties_MechShutter()`: Returns the properties of the mechanical shutter control for the selected Head. If hd_no is not specified, returns the properties for the currently active head. Properties include: IsSupported, min, max, one shot avail, auto avail. The one shot avail and auto avail modes are not applicable. **Note:** The "is_supported" value indicates if the shutter control capability is available in the camera feature set. It is not an indication if this particular head has a shutter present. Therefore, if the feature is available, the is_sup value will be 1, regardless of the camera head type.

NOTE: Mechanical shutters are present in Princeton Instruments' full frame Camera Heads, the ES 1602/1603 and ES 3200. The mechanical shutter functions have no effect on Camera Heads employing interline sensors.

Command String: "225 (hd_no)<LF>"
Response: "225 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max value (1)
<min> => min value (0)
<one_shot> = 0=not supported
<auto> = 0=not supported

`SetMechShutter(state, lockpos)`: Toggle whether the mechanical shutter is active or not. When the shutter is off, the lockpos argument determines whether the shutter is locked in the open or closed position. The amount of time that the shutter stays open during image acquisition is determined by the triggered integration time control. This function has no effect on camera heads without a mechanical shutter. If hd_no is not specified, sets the shutter state for the currently active head.
Command String: "226 <state> <lockpos) (hd_no)<LF>"
Where: <state> => 1=on, 0=off
<lockpos> => 1=open, 0=closed
Response: "226 <status> 0<LF>"
Where: 0 => return parameter count

`GetMechShutter(state, lockpos)`: Returns the current values for shutter state and lock position. If hd_no is not specified, returns the shutter state for the currently active head.
Command String: "227 (hd_no)<LF>"
Response: "227 <status> 2 <state> <lockpos><LF>"
Where: 2 => return parameter count
<state> = shutter state returned from camera
<lockpos> = lock position returned from camera

14.6.8 Defect Correction

`Properties_ConcealDefect`: Returns the properties of the defect concealment feature for the controller. Properties include: IsSupported, min, max, one shot, and auto. One shot and auto are not applicable and return a value of 0.
Command String: "235 (hd_no)<LF>"
Response: "235 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max value (1)
<min> => min value (0)
<one_shot => N/A: value = 0
<auto> => N/A: value = 0

`SetDefectConcealState (state)`: Toggle defect concealment for the specified camera head.
Command String: "236 <state> <hd_no> <LF>"
Where: <state> => 1 = correction on, 0 = correction off
<hd_no> => target camera head
Response: "236 <status> 0 <LF>"
Where: 0 => return parameter count

`GetDefectConcealState (state)`: Returns current state of defect concealment for the specified camera head.
Command String: "237 <state> <hd_no> <LF>"
Where: <state> => 1 = correction on, 0 = correction off
<hd_no> => target camera head
Response: "237 <status> 0 <LF>"
Where: 0 => return parameter count

`GetDefectListInfo (list_no)`: This command returns info on the specified defect list. Information returned includes: file present, camera head serial number, and 15 character comment field. The comment field can be a maximum of 15 characters long. Any comment longer than 15 characters will be truncated to 15 characters.
Command String: "238 <list_no> <LF>"
Where: <list_no> => allowable values: 1, 2, 3, or 4 (table index)
Response: "238 <status> 3 <present> <serno> <comment field> <LF>"
Where: 3 => return parameter count
<present> = Boolean value indicating if the specified normalization table is present in the camera
<serno> = serial number of the camera head this file applies to
<comment field> = maximum of 15 characters extracted from the comment field of the normalization table file

`LoadDefectList (table_no)`: This command causes the camera to load the specified defect list data from the camera's compact flash storage into the active memory for use in correcting image data. The command initiates the process, but returns before list loading is complete. If the specified flat field file is not found, an error status is returned. The table number specified must have the value 1, 2, 3, or 4. This index specifies which of the four lists that may be present the camera should be loaded. This index value is not related to a specific camera head in the camera.
Command String: "239 <table_no> <LF>"
Where: <table_no> => allowable values: 1, 2, 3, or 4 (table index)

14.6.9 Sensor Normalization

`Properties_FlatField (val)`: Returns the properties of the flat field normalization feature for the specified camera head. Properties include: IsSupported, min, max, one shot, and auto.
Command String: "242 (hd_no)<LF>"
Response: "242 <status> 5 <is_sup> <max> <min> <one_shot> <auto> <LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => N/A, returns 0
<min> => N/A, returns 0
<one_shot => 1 = one shot supported, 0 = not supported
<auto> => 1 = auto mode supported, 0 = not supported

`SetFFState (hdno, state)`: This command controls whether the flat field gain and offset correction for the currently selected head is active or not. When "on" all pixel values for the sensor are corrected with the values specified in the normalization tables. A separate correction table is maintained for each sensor.
Command String: "243 <bool> (hd_no)<LF>"
Where: <bool> => 1 = correction on, 0 = correction off
Response: "243 <status> 0 <LF>"
Where: 0 => return parameter count

`GetFFState (hdno, state)`: This command returns the current state of the flat field correction (on or off) for the specified head.
Command String: "244 <bool> (hd_no)<LF>"
Response: "244 <status> 1 <value> <LF>"
Where: 1 => return parameter count
<bool> = current state of smear correction function, 1= correction on, 0 = correction off

14.6.10 Binning

Important: Binning is not supported in Camera Heads with color sensors!

Binning is a technique whereby the signals from adjacent pixels in a CCD are combined to produce an effective array with larger pixels, lower resolution, and faster frame rates. Binning is supported with monochrome sensors only.

Caution: The *MEGAPLUS* software and firmware allow binning levels to be specified for a color camera head and the camera will apply binning for that head. However, it should be noted that the binning process is not color-aware and will not account for pixel color in the binning process. As a result, the colors in the binned image will be incorrect.

In the *MEGAPLUS* Four-Head Monochrome FPGA configuration, the current binning state applies to all heads. Setting the binning level for any head will apply the same binning state to all heads. In the Two-Head Advanced Color configuration, each head maintains its own binning state.

`Properties_Binning()`: Returns the properties of the binning feature for the specified Camera Head. Properties include: IsSupported, min, max, one shot, auto. One shot and auto are not applicable.
Command String: "247 (hd_no)<LF>"
Response: "247 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max value (4)
<min> => min value (1)
<one_shot> => N/A, value = 0
<auto> => N/A, value = 0

`SetBinning(binlevel)`: Sets the binning mode for the specified Camera Head. Important Note: In the *MEGAPLUS* Four-Head Monochrome FPGA configuration, the current binning state applies to all heads. Setting the binning level for any head will apply the same binning state to all heads. In the Two-Head Advanced Color configuration, each head maintains its own binning state.
Command String: "248 <binlevel> (hd_no)<LF>"
Where: <binlevel> => binning level
1 = 1x1 binning (i.e., no binning)
2 = 2x2 binning
3 = 3x3 binning
4 = 4x4 binning
Response: "248 <status> 0<LF>"
Where: 0 => return parameter count

`GetBinning(binlevel)`: Returns the binning level for the specified Camera Head.
Command String: "249 (hd_no)<LF>"
Response: "249 <status> 1 <binlevel><LF>"
Where: 1 => return parameter count
<binlevel> => binning level
1 = 1x1 binning (i.e., no binning)
2 = 2x2 binning
3 = 3x3 binning
4 = 4x4 binning

14.7 Controller Related Functions

The following functions apply to the camera as a whole, rather than to individual Camera Heads.

14.7.1 Trigger

`Properties_Trigger()`: Returns the properties of the trigger feature for the controller. Properties include: IsSupported, max, min, and polarity control.
Command String: "401 <LF>"
Response: "401 <status> 4 <is_sup> <max> <min> <polarity><LF>"
Where: 4 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max trigger mode number (7)
<min> => min trigger mode number (0)
<polarity> = 0=cannot specify trigger polarity
1=can specify trigger polarity

`SetTriggerPolarity(pol_value)`: This function sets the value for the trigger polarity for the specified camera. A positive polarity means the trigger signal will be considered active when the signal is in a high or true state. Negative polarity is the reverse - the trigger signal will be considered active when the signal is in a low state.
Command String: "402 <pol_value> (hd_no)<LF>"
Where: <pol_value> => 1 = positive logic
0 = negative logic
Response: "402 <status> 0<LF>"
Where: 0 => return parameter count

`GetTriggerPolarity(pol_value)`: Returns the current state of trigger polarity setting for the controller. A positive polarity means the trigger signal will be considered active when the signal is in a high or true state. Negative polarity is the reverse - the trigger signal will be considered active when the signal is in a low state.
Command String: "403<LF>"
Response: "403 <status> 1 <pol_value><LF>"
Where: 1 => return parameter count
<pol_value> = trigger polarity setting returned from the camera

`SetTriggerMode(mode, param)`: Sets the trigger mode for the controller. Mode number must be within the range specified by the Trigger Properties command. Note: In addition to setting the trigger mode, triggering must be enabled with the SetTriggerState() command in order for triggered operation to take effect. The count parameter is used for some trigger modes. If the count value is not required for a given trigger mode, its value will be ignored.
Command String: "404 <mode> <parm><LF>"
Where: <mode> =>trigger mode no.
0 = Edge triggering (asynchronous reset)
1 = Edge triggering with pulse width controlled integration
4 = Double exposure
6 = Periodic interval (internal self trigger)
7 = Overlapped edge mode
<parm> => parameter (This value only relevant to certain trigger modes, otherwise ignored.)
Response: "404 <status> 0<LF>"
Where: 0 => return parameter count

`GetTriggerMode(mode)`: Returns the current state of trigger mode number for the controller.
Command String: "405<LF>"
Response: "405 <status> 2 <mode> <parm><LF>"
Where: 2 => return parameter count
<mode> = trigger mode number returned from the camera
<parm> = parameter value for those modes that use a parameter, otherwise 0

`SetTriggerState(state)`: Turns triggering ON or OFF. When triggering is enabled (ON), the camera will recognize and respond to trigger events in the manner dictated by the trigger mode. When triggering is disabled, all trigger signals will be ignored and the camera will operate in a free-run video mode.
Command String: "406 <state> <LF>"
Where: <state> => 0 = Triggering OFF
1 = Triggering ON
Response: "406 <status> 0<LF>"
Where: 0 => return parameter count

`GetTriggerState(mode)`: Returns the current state of triggering.
Command String: "407<LF>"
Response: "407 <status> 1 <state> <LF>"
Where: 1 => return parameter count
<state> => 0 = Triggering OFF
1 = Triggering ON

`SetTriggerSource(source)`: Specifies the input source that will initiate the trigger event. Sources can include: external trigger (via BNC on controller rear panel), CameraLink trigger signal (CC1 in the CameraLink signal specification), or software trigger. The software trigger command only applies to mode 0 (Edge Trigger, Asynchronous Reset) and 7 (Overlapped Edge Mode). Use the SoftwareTrigger() command to cause a software trigger event.
Command String: "408 <source> <LF>"
Where: <source> => 0 = External trigger via BNC
1 = CameraLink trigger signal on CC1
2 = Software trigger command
Response: "408 <status> 0<LF>"
Where: 0 => return parameter count

`GetTriggerSource(source)`: Returns the current trigger source that will initiate a trigger event.
Command String: "409<LF>"
Response: "409 <status> 1 <source> <LF>"
Where: 1 => return parameter count
<source> => 0 = External trigger via BNC
1 = CameraLink trigger signal on CC1
2 = Software trigger command

`SetTriggerIntTime(itime)`: When operating in triggered modes that use a programmable integration time, this command specifies the integration time that will be used for the triggered exposure. This control offers a range of integration time values that can be beyond the range available with the standard free-run sensor integration time. The specified integration time will be applied to all Camera Heads when triggering is enabled and a trigger is received.

NOTE: In some trigger modes (for example, Mode 1) integration time is controlled by other means, such as pulse width. In this case the Triggered Integration Time value has no effect.

Important Things to Know

The Triggered Integration Time value applies to all attached camera heads. This value will be applied in trigger mode 0 (Edge Trigger, Asynchronous Reset), 6 (Periodic Interval), or 7 (Overlapped Edge Mode) active.

This function controls triggered integration time only. When the camera is operating in free-run (video) mode, integration time is controlled via the SetIntTime() command.

Triggered mode operation provides a larger range of integration times. Free-run integration time is limited to one frame-readout time at the current clock rate.

Command String: "410 <itime> <LF>"
Where: <itime> => triggered integration time. Unit is milliseconds. Specified as floating point with 3 digits of fractional precision (x.xxx) to provide 1 microsecond resolution. Range is 0.001 (1 µsec) to 10,000 msec (10 seconds) for trigger modes 0-6. For trigger mode 7, range is 0.010 (10 µsec) to 600,000 (10 minutes).
Response: "410 <status> 0<LF>"
Where: 0 => return parameter count

`GetTriggerIntTime(itime)`: Returns the currently specified value for triggered integration time in msec.
Command String: "411<LF>"
Response: "411 <status> 1 <itime> <LF>"
Where: 1 => return parameter count
<itime> => the current triggered integration time. Unit is milliseconds. Specified as floating point with 3 digits of fractional precision (x.xxx) to provide 1 microsecond resolution. Range is 0.001 (1 µsec) to 10,000 msec (10 seconds) for trigger modes 0-6. For trigger mode 7, range is 0.010 (10 µsec) to 600,000 (10 minutes.

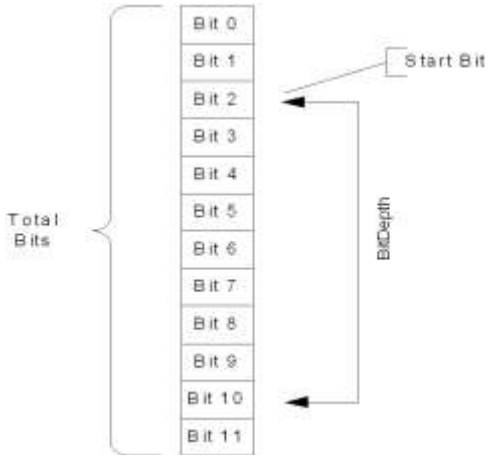
`SWTrigger()`: This command creates a trigger event via a camera command, rather than an external hardware trigger. When the camera receives this command, it will cause a trigger event for the current trigger mode. For the event to be acknowledged by the camera, the following conditions must be met. The camera should be configured for a trigger mode compatible with software triggering, the trigger source should be set to software, and triggering should be enabled. The software trigger command only applies to mode 0 (Edge Trigger, Asynchronous Reset) and 7 (Overlapped Edge Mode).
Command String: "412<LF>"
Response: "412 <status> 0 <LF>"
Where: 0 => no parameters returned

14.7.2 Bit Window

`Properties_BitWindow()`: Returns the properties of the bitdepth/bit window feature for the controller. This feature allows user-control of which of the available 12 bits of data per pixel are output from the camera. Properties include: IsSupported, min, max, one shot avail, auto avail.
Command String: "424 <LF>"
Response: "424 <status> 5 <is_sup> <max> <min> <one_shot><auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max bitdepth (12)
<min> => min bitdepth (8)
<one_shot> => N/A: value = 0
<auto> => N/A: value = 0

`SetBitWindow(bitdepth, LSB)`: This command specifies the bit depth and the bit window location for the camera. The camera's pixel data is processed at 12 bits internally, and can be output at 8, 10, or 12 bits per pixel. The bit depth argument of the BitWindow command specifies the system's output pixel depth. The LSB argument defines the least significant bit to be output. This allows examination of a subset of the available output dynamic range. The bit specified as the LSB will be assigned to bit 0 of the output data. The output data will include bits from the specified bits ranging from LSB to LSB+(bitdepth-1) of the internal data. If the bit depth of the window specified is less than the selected bit depth, the remaining most significant bits will be zero-filled.

The bit configuration specified by this command is the input data to any additional data formatting created by the output interface. For example the "system" bit depth can be set to a bit depth of 10 with this command which will cause the camera to output 10 bits of pixel data. However, the FireWire interface only supports data formats of 8-bit mono and 16-bit mono. In this case, the input 10-bit pixel data will be expanded to a16 bit format (10 bits of data plus 6 MSB zero bits) for compatibility with the data transfer protocol.



This provides a windowing function that enables the user to specify which of the available data bits are output. The bit number specified for LSB is zero-based– i.e., the least significant bit in an 8-bit word is bit 0 and the most significant bit is bit 7. ***Note***: The maximum bit depth setting is "8" if using a single-head controller and outputting "processed RGB data".

Command String: "425 <bitdepth> <LSB><LF>"
Where: <bitdepth> => the total number of bits to output per pixel (must be 8, 10, or 12). Maximum allowable value can be determined from the properties_Bitwindow command.
<LSB> => the zero-based bit number that will be the least significant bit of the output data. Range is 0-4 when bitdepth is 8, 0-2 when bitdepth is 10, and 0-0 when bitdepth is 12.
Response: "425 <status> 0<LF>"
Where: 0 => return parameter count

`GetBitWindow(bitdepth, LSB)`: Returns the current bit depth and bit window location values from the camera.
Command String: "426<LF>"
Response: "426 <status> 2 <bitdepth> <LSB><LF>"
Where: 2 => return parameter count
<bitdepth> => the total number of bits to output per pixel (must be 8, 10, or 12). Depth of 12 not supported in 10-bit cameras.
<LSB> => the zero-based bit number that should be the least significant bit of the output data.

14.7.3 Strobe Polarity

`Properties_Strobe(val)`: Returns the properties of the strobe feature for the controller. Properties include: IsSupported, min, max, one shot, and auto. The max, min, one shot, and auto modes are not applicable.
Command String: "433 <LF>"
Response: "433 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
< max> => N/A, returns 0
<min> => N/A, returns 0
<one_shot> => N/A, returns 0
<auto> => => N/A, returns 0

`SetStrobePolarity(val)`: Sets the output strobe signal polarity.
Command String: "434 <bool><LF>"
Where: <bool> =>1 = positive logic, 0 = negative logic
Response: "434 <status> 0<LF>"
Where: 0 => return parameter count

`GetStrobePolarity(val)`: Returns the output strobe signal polarity.
Command String: "435<LF>"
Response: "435 <status> 1 <polarity><LF>"
Where: 1 => return parameter count
<polarity> = current strobe polarity
1= positive logic
0 = negative logic

14.7.4 Strobe Delay

`SetStrobeDelay(val)`: Sets the delay between the time that the trigger signal is received and the time that the strobe signal is output. Delay is in milliseconds. If a delay of zero is specified, the strobe will be output immediately after receipt of the trigger. If a delay period is specified that is longer than the integration time of the sensor (determined either by programmed value or pulse width), the strobe will be output immediately at the end of the integration period.
Command String: "436 <delay><LF>"
Where: <delay> => strobe delay in milliseconds, Unit is milliseconds. Specified as floating point with 3 digits of fractional precision (x.xxx) to provide 1 microsecond resolution. Range 0 to 1,000 msec (1 second)
Response: "436 <status> 0<LF>"
Where: 0 => return parameter count

`GetStrobeDelay(val)`: Gets the current value for the delay between the time that the trigger signal is received and the time that the strobe signal is output. Delay is specified in milliseconds. If a delay period is specified that is longer than the integration time of the sensor (determined either by programmed value or pulse width), the strobe will be output immediately at the end of the integration period.
Command String: "437<LF>"
Response: "437 <status> 1 <delay><LF>"
Where: 1 => return parameter count
<delay> = current value of strobe delay. Unit is milliseconds. Specified as floating point with 3 digits of fractional precision (x.xxx) to provide 1 microsecond resolution. Range 0 to 1,000 msec (1 second).

14.7.5 Gamma

NOTE: The Gamma function uses the output LUTs to apply a gamma correction. When the color space LUTs are enabled by this function, gamma is effectively disabled (because the output LUTs are in use for this function). Generally LUTs employed in color space conversion and correction include a gamma correction, so the ability to include gamma correction is up to the user to control by determining the contents of the LUTs.

`Properties_Gamma()`: Returns the properties of the gamma feature camera. Gamma is a controller property and is applied as a LUT. In a color sensor, gamma is applied after the Bayer interpolation and at the output of the color conversion engine. Properties include: IsSupported, min, max, one shot avail, auto avail.
Command String: "438 <LF>"
Response: "438 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max gamma value
<min> => min gamma value
<one_shot> N/A: value = 0
<auto> => N/A: value = 0

`SetGammaValue(gamma_value)`: Sets the gamma value for the camera to the specified value. The gamma value passed is a floating point number with up to three decimal places formatted as an ASCII string (x.xxx). Typical values for gamma range between 1.000 and 1.800. The max/min range for this function will be specified by the Properties function. Values of gamma between 0.100 and 1.000 apply an exponential gamma which tends to stretch bright values in the image. Gamma values greater than 1.000 stretch the dark values in the image at the expense of the bright values. A value of 1.000 provides a linear gamma output, which is the same as disabling gamma correction.

Use GetGammaState to enable and disable gamma correction. In order for gamma processing to take effect, the gamma value must be set AND the gamma state must be enabled (ON).
Command String: "439 <value><LF>"
Where: <value> => the gamma. Must fall within min and max specified in Properties_Gamma command.
Response: "439 <status> 0<LF>"
Where: 0 => return parameter count

`GetGammaValue(gamma_value)`: Returns the current gamma value for the camera. The gamma value passed is a floating point number with up to three decimal places formatted as an ASCII string.
Command String: "440<LF>"
Response: "440 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> = gamma value returned from the camera.

`SetGammaState(gamma_value)`: Enables or disables gamma correction. Before enabling gamma processing, you should set the desired gamma value using the SetGammaValue() command.
Command String: "441 <value><LF>"
Where: <value> =>1 = gamma on, 0 = gamma off
Response: "441 <status> 0<LF>"
Where: 0 => return parameter count

`GetGammaState(gamma_value)`: Returns the current state of the gamma correction function - enabled or disabled.
Command String: "442<LF>"
Response: "442 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> = gamma state returned from the camera.

14.7.6 Color Space Conversion

NOTE: In the Dual Head Advanced Color FPGA configuration, the gamma factor is applied via the output LUTs for the color space conversion engine. If other look up tables are downloaded to the camera, the gamma factor will have no effect.

e

`Properties_ColorSpaceTransform()`: Returns the properties of the ColorSpaceTransform function. This function supports a color space conversion engine that provides 3 input lookup tables, a 3 x 3 matrix conversion, and 3 output lookup tables. This conversion engine meets the calculation for a variety of color space conversions and corrections. Properties include: IsSupported, min, max, one shot avail, auto avail. Max and Min refer to the on/off property. One shot and auto are not applicable and return a value of 0.
Command String: "443<LF>"
Response: "443 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max value: 1 = enabled
<min> => min value: 0 = disabled
<one_shot> N/A: value = 0
<auto> => N/A: value = 0

`SetColorSpaceMatrixCoefficient()`: Specifies the nine coefficient values for the color space conversion matrix. Coefficients are passed as an ASCII representation of floating point values in a format of xxx.nnn where x=integer places and n=decimal values. An optional sign bit may be specified (for example, +/-999.999). Coefficients are specified in a row dominant order – i.e., values for the first row (0,0) (0,1) (0,2), then the second row (1,0) (1,1) (1,2) and the third row (2,0) (2,1) (2,2).

Command String: "444 <c1> <c2> <c3> <c4> <c5> <c6> <c7> <c8><c9><LF>"
Where: <c1>..<c9> = color space matrix coefficient values (xx.nnn format)
Response: "444 <status> 0<LF>"
Where: 0 => return parameter count

`GetColorSpaceMatrixCoefficient()`: Returns the nine coefficient values for the color space conversion matrix. Coefficients are passed as an ASCII representation of floating point values in a format of xxx.nnn where x=integer places and n=decimal values. An optional sign bit may be specified (for example, +/-999.999). Coefficients are specified in a row dominant order i.e. values for the first row (0,0) (0,1) (0,2), then the second row (1,0) (1,1) (1,2) and the third row (2,0) (2,1) (2,2).
Command String: "445<LF>"
Response: "445 <status> 9 <c1> <c2> <c3> <c4> <c5> <c6> <c7> <c8> <c9><LF>"
Where: 9 => return parameter count
<c1>…<c9> = color space matrix coefficient values

`SetColorSpaceTransformState(state)`: Enables or disables application of the 3x3 color space conversion matrix processing for the specified camera. The matrix coefficients must be set to the desired values using the SetColorSpaceMatrixCoefficient() command. Enabling the transform processing causes the matrix of coefficients to be applied to the color pixel data.
Command String: "446 <value><LF>"
Where: <value> =>1 = conversion on, 0 = conversion off
Response: "446 <status> 0<LF>"
Where: 0 => return parameter count

`GetColorSpaceTransformState(state)`: Returns the current state of the 3x3 color space conversion matrix processing in the camera: enabled or disabled.
Command String: "447<LF>"
Response: "447 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> = conversion state returned from the camera.

`SetColorSpaceLUTState(state)`: Enables and disables application of the three input lookup tables (LUTs) and the three output lookup tables (LUTs) that are a part of the color space conversion/correction engine. When the color space LUTs are enabled, the pixel values from the sensor are translated by the LUTs. When the color space LUTs are disabled, the pixel values from the sensor are not affected by the LUTs.

The LUTs must be downloaded to the camera in a manner similar to defect lists and normalization tables. If no LUTs have been downloaded, the conversion processing has no effect on the image data.

NOTE: Color space coefficient values are controlled by this function. However, to have them take effect the color space transformation must be enabled with the SetColorSpaceTransformState command.

IMPORTANT NOTE: The Gamma function uses the output LUTs to apply a gamma correction. When the color space LUTs are enabled by this function, gamma is effectively disabled (because the output LUTs are in use for this function). Generally, LUTs employed in color space conversion and correction include a

gamma correction so the ability to include gamma correction becomes the user's control by determining the contents of the LUTs.

Command String: "465 <value><LF>"
Where: <value> =>1 = conversion on, 0 = conversion off
Response: "465 <status> 0<LF>"
Where: 0 => return parameter count

`GetColorSpaceLUTState(state)`: Returns the current state of the color space lookup table processing - enabled or disabled.
Command String: "466<LF>"
Response: "466 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> = conversion state returned from the camera.

14.7.7 Sensor Tap Readout

`Properties_SensorTapReadout()`: Returns the properties of the Sensor Tap Readout feature. This feature allows users to select between single tap and dual tap sensor readout.

NOTE: Sensor tap readout configuration does not affect the camera's data output configuration. When a sensor is read in dual tap mode, the two taps are interleaved in the camera back into a single image stream. Not all camera heads will support dual tap readout. The *MEGAPLUS* full-frame heads including the ES 1602[*]/3 and ES 3200 do not support dual tap operation.

Command String: "448<LF>"
Response: "448 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max value: (2)
<min> => min value: (1)
<one_shot> N/A: value = 0
<auto> => N/A: value = 0

`SetSensorTapReadout()`: Specifies mode to read pixel data from the sensor. Currently supported mode selections include single tap and dual tap. This mode applies to all sensors attached to the camera. If the sensors in the Camera Heads do not support dual tap operation, the readout will be single tap and this command will have no effect on operation. Not all camera heads will support dual tap readout. The *MEGAPLUS* full-frame heads including the ES 1602[*]/3 and ES 3200 do not support dual tap operation. The effect of this function in custom mixed-sensor configurations will be defined per application.
Command String: "449 <readout_mode><LF>"
Where: <readout_mode> => 1 = single tap, 2 = dual tap
Response: "449 <status> 0<LF>"
Where: 0 => return parameter count

[*] ES 1602 has been discontinued.
[*] ES 1602 has been discontinued.

`GetSensorTapReadout()`: Returns the current value of the sensor tap readout mode. This mode applies to all sensors attached o the camera.
Command String: "450<LF>"
Response: "450 <status> 1 <readout_mode><LF>"
Where: 1 => return parameter count
<readout_mode> => 1 = single tap, 2 = dual tap

14.7.8 Video Output (Multi-head controller only)

`Properties_VideoOutput()`: Returns the properties of the video output function. Properties include: IsSupported, min, max, one shot, auto. If the video output option is not available in this camera, the function will return is_supported = 0. One shot and auto are not applicable and return a value of 0.
Command String: "451<LF>"
Response: "451 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => 1
<min> => 0
<one_shot> N/A: value = 0
<auto> => N/A: value = 0

`SetVideoState (state)`: This command controls whether the video output option is enabled. When video output is "on" the digital image data is converted to a video signal that is available at the DVI connector on the controller rear panel. When video output is "off" the video conversion circuitry is powered down, reducing power requirements and heat generation in the controller.
Command String: "452 <bool><LF>"
Where: <bool> => 1 = video on, 0 = video off
Response: "452 <status> 0<LF>"
Where: 0 => return parameter count

`GetVideoState(state)`: Returns the current value of the video output (on or off).
Command String: "453<LF>"
Response: "453 <status> 1 <value><LF>"
Where: 1 => return parameter count
<value> = current state of the video output feature (1 = video on, 0 = video off)

14.7.9 Settings Memory

`Properties_MemSettings()`: Returns the properties of the save settings to memory function. Properties include: IsSupported, max, min, one shot, and auto. The min and max values specify the number of settings groups that can be saved in the camera. Set number 0 is reserved for factory default settings. One shot and auto are not applicable and return a value of 0.
Command String: "454 <LF>"
Response: "454 <status>5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max set no = 15
<min> => min set no = 1 (0 is reserved for factory default)
<one_shot = 0=not supported
<auto> = 0=not supported

`SaveMemSettings(mem_setno)`: Save the current camera modes and settings to internal memory under the specified set number. Setno value cannot be 0. The value 0 is reserved for factory settings.
Command String: "455 <set_no><LF>"
Where: <set_no> => memory set number
Response: "455 <status> 0<LF>"
Where: 0 => return parameter count

`ReadMemSettings(mem_setno)`: Loads the specified set of saved camera modes and settings from internal memory. If setno = 0, the factory default settings will be restored.
Command String: "456<LF>"
Response: "456 <status> 0<LF>"
Where: 0 => return parameter count

14.7.10 Controller Temperature

`Properties_ConsoleTemp`: Returns the properties of the controller temperature feature. Properties include: IsSupported, min, max, one shot, and auto. The only property applicable to this feature is the IsSupported property. All others are not applicable and return a value of 0.
Command String: "457 (hd_no) <LF>"
Response: "457 <status> 5 <is_sup> <max> <min> <one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => N/A: value = 0
<min> => N/A: value = 0
<one_shot> N/A: value = 0
<auto> => N/A: value = 0

`GetConsoleTemp (tTop, tBottom, tPS)`: This command returns three health-monitoring temperatures from the camera controller. Values returned are the temperature of the top of the controller processor board, the bottom of the processor board, and the power supply temperature. All values are in degrees Celsius.
Command String: "458<LF>"
Response: "458 <status> 1 <tTop> <tBottom> <tPS> <LF>"
Where: 1 => return parameter count
<tTop> = temperature from the top of the controller processor board
<tBottom> = temperature from the bottom of the controller processor board
<tPS> = temperature of the controller power supply

NOTE: For the single-head controller, the bottom and power supply temperature readings share the same sensor.

14.7.11 Quick Mux

`SetQuickMux(outputSel, mode)`: This function configures the camera data paths to output image data from a specific Camera Head or in the case of the Four-Head Monochrome CameraLink configuration, from a specific combination of camera heads. The format of the resulting pixel data stream is a function of the current system bit window setting and the head spectral type (mono/color). The Mode parameter controls whether the data from a color sensor is output in raw or processed form.
Command String: "459 <outputSel> <mode> <LF>"
Where: <outputSel> => the imaging source selected for output. Values are 1, 2, 3, or 4 (Head 1 - Head 4). A value of 99 indicates a custom output configuration (only available in Four Head Mono configuration for CameraLink output).
<mode> => ignored for monochrome sensors. For Bayer color sensors, 0 = processed RGB data; 1 = raw Bayer data
Response: "459 <status> 0<LF>"
Where: 0 => return parameter count

NOTE: For the single-head controller, outputSel is limited to 1.

`GetQuickMux(outputSel, mode)`: Returns the current outputSel and mode settings for the QuickMux feature.
Command String: "460<LF>"
Response: "460 <status> 2 <outputSel> <mode><LF>"
Where: 2 => return parameter count
<outputSel > => the current value for the outputSel setting for this feature. Values are 1, 2, 3, or 4 (Head 1 - Head 4). A value of 99 indicates a custom output configuration (only available in Four Head Mono configuration for CameraLink outputs).
<mode> => Value = N/A for mono sensors (value=0). For color sensors, 0 = processed RGB data out, 1 = raw Bayer data out

14.7.12 CameraLink Output Multiplexing (CLMux)

(Multi-head controller only)

When the CameraLink output is used as the image transfer mode, the CameraLink multiplexing function can be used to control which image data is output at each of the available CameraLink data ports.

NOTE: The *CLConfigClass* commands are not supported in the 1394 control protocol and have no effect on 1394 camera operation.

`SetCLMux(P1Mux, P2Mux, P3Mux, P4Mux, P5Mux, P6Mux)`: The *MEGAPLUS* platform supports Medium format and Dual Base CameraLink output. Depending on the selected bit depth and frame grabber configuration, there may be up to four output taps or ports to send image data. This command provides a means to map the image planes of data available within the camera to the specified output tap on the CameraLink output. The number of available output ports will be dependent on the current bit depth setting, the current Quick Mux setting, the CameraLink class configuration setting (Medium or Dual Base), and the physical attributes of the specific frame grabber (Base, Medium, or Dual Base). The use of the CameraLink Mux command varies depending on the FPGA configuration loaded.

The CLMux commands are closely tied to the QuickMux commands. The QuickMux commands provide a quick and easy way to select an output configuration that outputs data from a single specified Camera Head. The CLMux commands give more control of special customized output configurations**. In order for the configuration selected by**

the SetCLMux command to take effect, the SetQuickMux selection must be set to "Custom" (value = 99).

Dual-Head Color FPGA Configuration: The assignment of Camera Head output to CameraLink tap is predetermined in this configuration. This command has no effect in the Dual-Head Color FPGA configuration.

Four-Head Monochrome FPGA Configuration: The CameraLink Mux commands can be used to control assignment of pixel streams from any of the four monochrome Camera Heads to CameraLink output tap.

NOTE: This command has no effect in the Dual-Head Color FPGA configuration.

In the command format below, the available image planes are identified as follows:

1 - Image Data from Sensor 1 (mono or raw Bayer)
2 - Image Data from Sensor 2 (mono or raw Bayer)
3 - Image Data from Sensor 3 (mono or raw Bayer)
4 - Image Data from Sensor 4 (mono or raw Bayer)

Command String: "461 <P1mux> P2mux><P3mux><P4mux><P5mux><P6mux><LF>"
Where: <P1mux> = Image plane specification for Port 1
<P2mux> = Image plane specification for Port 2
<P3mux> = Image plane specification for Port 3
<P4mux> = Image plane specification for Port 4
<P5mux> = <Reserved for future use>
<P6mux> = <Reserved for future use>
Response: "461 <status> 0<LF>"
Where: 0 => return parameter count

`GetCLMux(P1Mux, P2Mux, P3Mux, P4Mux, P5Mux, P6Mux)`: Returns the current values for the CLMux assignments. The values returned identify the available image planes as follows:

0 - No data. Channel is turned off to output only zeros.
1 - Image Data from Sensor 1 (mono or raw Bayer)
2 - Image Data from Sensor 2 (mono or raw Bayer)
3 - Image Data from Sensor 3 (mono or raw Bayer)
4 - Image Data from Sensor 4 (mono or raw Bayer)

Command String: "462 <LF>"
Response: "462<status>6<P1mux><P2mux><P3mux><P4mux><P5mux><P6mux> <LF>"
Where: 6 => return parameter count
<P1mux> = Image plane specification for Port 1
<P2mux> = Image plane specification for Port 2
<P3mux> = Image plane specification for Port 3
<P4mux> = Image plane specification for Port 4
<P5mux> = <Reserved for future use>
<P6mux> = <Reserved for future use>

14.7.13 Controller Reset

`ControllerReset()`: This function causes the controller to reinitialize itself. The initialization process is a "soft boot" process similar to that performed when the controller power is cycled. Command is executed immediately. No response is returned.
Command String: "463<LF>"

14.7.14 FireWire Reset (Multi-head controller only)

`FireWireReset()`: This function causes the controller to issue a 1394 bus Reset command. This issues a Reset command on the 1394 data bus which causes all devices on the bus to report a self-identification message.
Command String: "464<LF>"
Response: "464 <status> 0<LF>"
Where: 0 => return parameter count

14.7.15 Double Exposure TPD

`Properties_DoubleExpTPD()`: Returns the properties of the transfer pulse delay parameter used in double exposure trigger mode (Mode 4).
Command String: "467 <LF>"
Response: "467 <status> 5 <is_sup> <max> <min><one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max transfer pulse delay in 0.1 µs increments
<min> => min transfer pulse delay in 0.1 µs increments
<one_shot> => 0=not supported
<auto> => 0=not supported

`SetDoubleExpTPD()`: This command sets the transfer pulse delay for double exposure trigger mode (Mode 4). The allowable range is 0.1 to 10,000 microseconds in 0.1 microsecond intervals. However, due to CCD timing constraints, the minimum value that the CCD will actually implement depends on the camera model. For the minimum integration time for your camera, refer to the appropriate camera specification table in Section 18.4, "Camera Component Specifications", starting on page 162.
Command String: "468 <TPD value><LF>"
Where: <TPD> => the TPD value to be set
Response: "468 <status> 0 <LF>"
Where: 0 => no parameters returned

`GetDoubleExpTPD()`: This command returns the current transfer pulse delay for double exposure trigger mode. The allowable range is 0.1 to 10,000 microseconds in 0.1 microsecond intervals.
Command String: "469<LF>"
Response: "469 <status> 1 <TPD value><LF>"
Where: 1 => no parameters returned
<TPD value> = TPD value in 0.1 µs increments returned from the camera

14.7.16 Mode 6 Periodic Trigger Interval

`Properties_Mode6Interval()`: Returns the properties of the Mode 6, periodic trigger interval. This mode triggers the camera repeatedly on a period specified by the user.
Command String: "473 <LF>"
Response: "473 <status> 5 <is_sup> <max> <min><one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => max period in milliseconds (32,767)
<min> => min period in milliseconds (1)
<one_shot> => 0=not supported
<auto> => 0=not supported

`SetTriggerMode6Interval(period)`: When operating in triggered Mode 6 (Periodic Interval), which causes the camera to self-trigger an image repeatedly at a specified time interval, this function is used to specify the time period between triggers. The value is an integer number of milliseconds and must fall within the range returned from the Properties_Mode6Interval command. The typical range is 1/Max frame rate of the sensor to 5 minutes.
Command String: "474 <period> <LF>"
Where: <period> => the period in milliseconds
Response: "474 <status> 0<LF>"
Where: 0 => return parameter count

`GetTriggerMode6Interval( period)`: Returns the currently specified period value for trigger Mode 6.
Command String: "475<LF>"
Response: "475 <status> 1 <period> <LF>"
Where: 1 => return parameter count
<period> => the current value for this parameter

14.7.17 Pixel Clock Speed and Frequency

`Properties_CamPixClkSpeed()`: Returns the properties of the function to select pixel clock speed. This feature allows users to select between two different clocking speeds for sensor readout. Operating a sensor at higher clock speeds provides higher frame rates, but increased noise. Slower readout decreases the throughput, but minimizes noise in the image data. The feature allows users to optimize camera operation for the requirements of their application. Clock "speeds" (i.e., high/low) are selected rather than specific clock frequencies, which are sensor dependent. Properties include: IsSupported, min, max, one shot avail, auto avail. The one shot avail and auto avail modes are not applicable.
Command String: "479 <LF>"
Response: "479 <status> 5 <is_sup> <max> <min><one shot><auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => maximum speed selection value (1)
<min> => minimum speed selection value (0)
<one_shot = 0=not supported
<auto> = 0=not supported

`Set_CamPixClkSpeed(speed):` The camera provides user-selection of clock speeds. The camera supports more than one pixel clock selection. The camera provides the user with the option to run the sensors at different speeds to accommodate variations in frame grabbers or application requirements. The actual pixel clock frequency can be determined with the *Get_PixClkFrequency()* command. Sets the current pixel clock speed selection for the specified camera head.

NOTES:
1. **This feature is only active for configurations where all attached heads are identical. Configurations that utilize mixed heads only support the factory default pixel clock rate in order to accommodate the requirements of the unlike sensors.**
2. **Changing pixel clock speed causes the camera to perform initialization of the timing generators and several associated functions for the new speed. For this reason, it takes a few seconds for the camera to process the command and respond.**
3. **Changing the pixel clock speed while a frame grabber is grabbing live data may cause the frame grabber to timeout.**

Command String: "480 <speed> <LF>"
Where: <speed> => 1 = low-speed selection, 0 = high speed selection
Response: "480 <status> 0<LF>"
Where: 0 => return parameter count

NOTE: Some camera heads do not support dual-tap channel balance for two different clock speeds. If this is true of your camera, this command will change the clock speed, but may return a warning status. At higher some clock rates, you may see imbalance in the image when running in dual tap mode. Check with your Princeton Instruments application engineer to see if your camera can be calibrated for two operation speeds.

`Get_CamPixClkSpeed(speed)`: Returns the current pixel clock speed selection for the specified camera head. The actual pixel clock frequency can be determined with the *Get_PixClkFrequency()* command. Sets the current pixel clock speed selection for the specified camera head.
Command String: "481 <hd_no> <LF>"
Response: "481 <status> 0<speed><LF>"
Where: 1 => return parameter count
<speed> => 1 = low speed selection, 0 = high speed selection

14.7.18 CameraLink Config Class

When the CameraLink output is used as the image transfer mode, the CameraLink multiplexing function can be used to control which image data is output at each of the available CameraLink data ports. The combined effect of the Bit Window, CameraLink configuration class, Quick Mux and CameraLink multiplexing selections determine the configuration of data to the CameraLink interface.

NOTE: The *CLConfigClass* commands are not supported in the 1394a control protocol and have no effect on 1394a camera operation.

`SetCLConfigClass(Class)`: Specifies the CameraLink configuration to be activated. *MEGAPLUS* supports Base, Medium, and Dual Base CameraLink configuration classes. In addition, *MEGAPLUS* can support Alternating Tap output in the Dual Base configuration. Alternating tap divides the output image pixel stream into two parallel streams, one consisting of the odd numbered pixels and the other consisting of the even numbered pixels. This allows users to operate camera heads at their maximum frame rate at half the pixel clock rate required for single tap output. The signal assignments for Dual Base are different than those used in Medium and Base configurations. This function selects which signal output configuration will be used.

In Alternating Tap, the odd-numbered pixels of each row are output on one tap while the even-numbered pixels are output on the second tap. In this configuration, output of data from a single sensor requires two CameraLink "taps" or "ports". Therefore, it is possible to output image from a maximum of two heads in parallel.

NOTE: The camera must be operating in dual tap readout mode (*SetSensorTapReadout(2)*) before switching into Alternating Tap output mode.

Command String: "485 <CLClass><LF>"
Where: <CLClass> => 0 = Base/Medium
1 = Dual Base
2 = Dual Base with Alternating Tap
Response: "485 <status> 0<LF>"
Where: 0 => return parameter count

`GetCLConfigClass(CLClass)`: Returns the current value for the CameraLink configuration class.
Command String: "486 <LF>"
Response: "486 <status> <CLClass><LF>"
Where: 1 => return parameter count
<CLClass> => 0 = Base/Medium
1 = Dual Base
2 = Dual Base, Alternating Tap

14.8 Sensor-Related Functions

NOTE: For all functions that set parameters that apply to a lower level object than an camera head, the destination must be explicitly specified in the command. A complete target specification must specify the camera head, the sensor, and the tap. The head, sensor, and tap must be explicitly defined for each command.

14.8.1 CCD Analog Gain – Red

`Properties_CCDAnalogGainRed()`: Returns the properties of the CCD Analog Gain Red feature for the selected Head. Properties include: IsSupported, min, max, one shot, and auto. Although the analog gain function applies to a specific sensor/tap location, the assumption is made that the support, max, and min functions will be the same for all sensors and taps on a single head. Therefore it is only necessary to query the properties for a specific head. One shot and auto are not applicable and return a value of 0.
Command String: "9201 (hd_no) <LF>"
Response: "9201 <status> 5 <is_sup> <max> <min><one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => maximum value: 1 = enabled
<min> => minimum value: 0 = disabled
<one_shot> => N/A: value = 0
<auto> => N/A: value = 0

`SetCCDAnalogGainRed()`: Sets the analog gain value for the specified sensor (or tap where appropriate).
Command String: "9202 <value> <head> <sensor> <tap> <LF>"
Where: <value> => analog gain value
<head> <sensor> <tap> => target affected by setting
Response: "9202 <status> 0 <LF>"
Where: 0 => return parameter count

`GetCCDAnalogGainRed()`: Gets the analog gain value for the specified sensor (or tap where appropriate).
Command String: "9203 <head> <sensor> <tap> <LF>"
Where: <head> <sensor> <tap> => target affected by setting
Response: "9203 <status> 1 <value> <LF>"
Where: 1 => return parameter count
<value> = analog gain value

14.8.2 CCD Analog Gain – Green

`Properties_CCDAnalogGainGreen()`: Returns the properties of the CCD Analog Gain Green feature for the selected Head. Properties include: IsSupported, min, max, one shot, and auto. Although the analog gain function applies to a specific sensor/tap location, the assumption is made that the support, max, and min functions will be the same for all sensors and taps on a single head. Therefore, it is only necessary to query the properties for a specific head. One shot and auto are not applicable and return a value of 0.
Command String: "9204 (hd_no) <LF>"
Response: "9204 <status> 5 <is_sup> <max> <min><one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => maximum value: 1 = enabled
<min> => minimum value: 0 = disabled
<one_shot> => N/A: value = 0
<auto> => N/A: value = 0

`SetCCDAnalogGainGreen()`: Sets the analog gain value for the specified sensor (or tap where appropriate).
Command String: "9205 <value> <head> <sensor> <tap> <LF>"
Where: <value> => analog gain value
<head> <sensor> <tap> => target affected by setting
Response: "9205 <status> 0 <LF>"
Where: 0 => return parameter count

`GetCCDAnalogGainGreen()`: Gets the analog gain value for the specified sensor (or tap where appropriate).
Command String: "9206 <head> <sensor> <tap> <LF>"
Where: <head> <sensor> <tap> => target of request
Response: "9206 <status> 1 <value> <LF>"
Where: 1 => return parameter count
<value> = analog gain value

14.8.3 CCD Analog Gain – Blue

`Properties_CCDAnalogGainBlue()`: Returns the properties of the CCD Analog Gain Blue feature for the selected Head. Properties include: IsSupported, min, max, one shot, and auto. Although the analog gain function applies to a specific sensor/tap location, the assumption is made that the support, max, and min functions will be the same for all sensors and taps on a single head. Therefore, it is only necessary to query the properties for a specific head. One shot and auto are not applicable and return a value of 0.
Command String: "9207 (hd_no) <LF>"
Response: "9207 <status> 5 <is_sup> <max> <min><one_shot> <auto><LF>"
Where: 5 => return parameter count
<is_sup> => 1 = is supported, 0 = not supported
<max> => maximum value: 1 = enabled
<min> => minimum value: 0 = disabled
<one_shot> => N/A: value = 0
<auto> => N/A: value = 0

`SetCCDAnalogGainBlue()`: Sets the analog gain value for the specified sensor (or tap where appropriate).
Command String: "9208 <value> <head> <sensor> <tap> <LF>"
Where: <value> => analog gain value
<head> <sensor> <tap> => target affected by setting
Response: "9208 <status> 0 <LF>"
Where: 0 => return parameter count

`GetCCDAnalogGainBlue()`: Gets the analog gain value for the specified sensor (or tap where appropriate).
Command String: "9209 <head> <sensor> <tap> <LF>"
Where: <head> <sensor> <tap> => target of request
Response: "9209 <status> 1 <value> <LF>"
Where: 1 => return parameter count
<value> = analog gain value

15. GigE Vision Features

15.1 Introduction

15.1.1 GigE Vision™ Features Nomenclature and Structure

Typically, GigE Vision features are organized in a hierarchy of complexity: simple, intermediate, advanced, and invisible. Since the exact contents of a level may vary from application to application, this information is not included in the descriptions of the features. As a quick reference, the definition of the each level is provided below:

5. Simple (beginner) - features that should be visible for **all** users via the GUI and API. This is the default visibility in the XML files and will be used if the <Visibility> element is omitted. The number of features with "beginner" visibility should be limited to all **basic** features of the devices so the GUI display is well-arranged and is easy to use.
6. Intermediate (expert) - features that require a more in-depth knowledge of the camera functionality. This is the preferred visibility level for all advanced features in the cameras.
7. Advanced (guru) – advanced features that might bring the cameras into a state where it will not work properly anymore if it is set incorrectly for the cameras current mode of operation.
8. Invisible – features that should be kept hidden for the GUI users but still be available via the API.

In this document, features are grouped in the same manner as presented on the National Instruments® Measurement and Automation Explorer (MAX) Camera Attributes tab.

15.2 Camera Attributes

15.2.1 Acquisition and Trigger Controls

Acquisition and Trigger Controls lists all features that relate to actual image acquisition, including the triggering mode. Integration time is the amount of time the sensor collects photons for a single acquisition.

Important Things to Know

MEGAPLUS offers two integration time controls: one for free-run integration time and one for triggered operation. If you are running the camera in triggered mode, you must use the **`TriggerIntTime`** feature to control integration time.

The range of integration time values available is determined by 1) tap readout configuration, 2) pixel clock speed, and binning level. Any time that you change any of these parameters, the maximum and minimum free-run integration time values will also change. Therefore, whenever you change readout (**`SensorTapReadout)`**, speed (**`CamPixClkSpeed`**), or binning level (**`Binning)`**, the **`ExposureTimeMax`** feature will automatically go out and get the new maximum value.

AcquisitionFrameCount: (R/W) Not available for MEGAPLUS cameras.

AcquisitionMode: (R/W) This feature controls the acquisition mode of the device. It defines mainly the number of frames to capture during an acquisition and the way the acquisition stops. **AcquisitionMode** can take any of the following values:

- **SingleFrame:** One frame is captured.
- **MultiFrame:** Not available for MEGAPLUS cameras.
- **Continuous:** Frames are captured continuously until stopped with the **AcquisitionStop** command.

ExposureTimeAbs: (R/W) This feature is used to set the Exposure time (in microseconds) when **ExposureMode** is **Timed**. This controls the duration where the photosensitive cells are exposed to light.

TriggerSoftware: (Write) This feature is used to generate an internal trigger when **TriggerSource** is set to **Software.**

15.2.2 Analog Controls

GainAbs: (R/W) This feature controls the selected gain as an absolute physical value. This is an amplification factor applied to the video signal. Min: 0, Max: 36.

Gamma: (R/W) This feature is used to perform gamma correction of pixel intensity. This is typically used to compensate for non-linearity of the display system (such as CRT). Min: 0.10000000000000001, Max: 5.

NOTE: The Gamma function uses the output LUTs to apply a gamma correction. When the color space LUTs are enabled by this function, gamma is effectively disabled (because the output LUTs are in use for this function). Generally LUTs employed in color space conversion and correction include a gamma correction, so the ability to include gamma correction is up to the user to control by determining the contents of the LUTs.

GammaState: (R/W) This feature enables/disables gamma correction. Checked/Unchecked

15.2.3 Custom Features

ActiveHead: (R/W) This feature sets/returns the camera to be designated as the currently active camera head. Min: 1, Max:1

Binning: (R/W) This feature set/returns the binning to be used. Min: 1 (1x1 binning), Max: 4 (4x4 binning).

NOTE: Binning is not supported for color sensors.

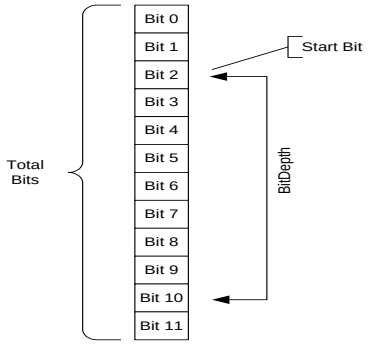
Binning is a technique whereby the signals from adjacent pixels in a CCD are combined to produce an effective array with larger pixels, lower resolution, and faster frame rates. Binning is supported with monochrome sensors only.

Caution: The *MegaPlus* software and firmware allow binning levels to be specified for a color camera head and the camera will apply binning for that head. However, it should be noted that the binning process is not color-aware and will not account for pixel color in the binning process. As a result, the colors in the binned image will be incorrect.

BitWindow: (R/W) This feature sets/returns the available data bits to be output and the LSB position. The camera's pixel data is processed at 12 bits internally, and can be output at 8, 10, or 12 bits per pixel depending on the LSB positioning. The LSB defines the least significant bit to be output. This allows examination of a subset of the available output dynamic range. The bit specified as the LSB will be assigned to bit 0 of the output data. The output data will include bits from the specified bits ranging from LSB to LSB + (bitdepth-1) of the internal data. If the bitdepth of the window specified is less than the selected bit depth, the remaining most significant bits will be zero-filled. Min: 2048, Max: 3072

NOTE: Determined by data output formatting.

The bit configuration specified by this command is the input data to any additional output data formatting created by the output interface. For example, the "system" bit depth can be set to a bit depth of 10 with this command which will cause the camera to output 10 bits of pixel data.



This provides a windowing function that enables the user to specify which of the available data bits are output. The bit number specified for LSB is zero-based – i.e., the least significant bit in an 8-bit word is bit 0 and the most significant bit is bit 7.

NOTES:
1. **The maximum bit depth setting is "8" if using a single-head controller and outputting "processed RGB data".**
2. **The data is compressed into an integer and is broken into two characters. The leftmost 8 bits express the bit depth in binary (8, 10, or 12) and the rightmost 8 bits express the LSB position of the bit depth.**
   Examples: 0x0804 represents 8 bits LSB in position 4
   0x0A01 represents 10 bits LSB in position 1

BlueGain: (**R/W) This feature sets/returns the blue** analog gain value for the camera (or tap where appropriate). The sensor tap is set by the SensorTapReadout feature. Min: 1, Max: 63

Brightness: (R/W) Reads/writes the brightness or offset for the currently active Head to the specified value in DN counts. Min: 1, Max: 255

CamPixClkFrequency: (Read) Returns the current pixel clock frequency in MHz for the active camera. Min: 0, Max: 40

CamPixClkSpeed: (R/W) The camera provides user-selection of clock speeds. The camera supports more than one pixel clock selection. The camera provides the user with the option to run the sensors at different speeds to accommodate variations in framegrabbers or application requirements. The low speed setting operates the sensor at a reduced rate (nominally 50% of maximum) to achieve slower, but lower noise operation. The high speed setting operates the sensor at its maximum clock speed for maximum throughput. The actual pixel clock frequency can be determined with the CamPixClkFrequency. Low/High

NOTES:
1. **Changing pixel clock speed causes the camera to perform initialization of the timing generators and several associated functions for the new speed. For this reason, it takes a few seconds for the camera to process the command and respond.**
2. **Changing the pixel clock speed while live data is being arquired may cause a timeout.**
3. **Some cameras do not support dual-tap channel balance for two different clock speeds. If this is true of your camera head, this command will change the clock speed, but may return a warning status. At higher some clock rates, you may see imbalance in the image when running in dual tap mode. Check with your Princeton Instruments application engineer to see if your camera can be calibrated for two operation speeds.**

ConsoleTempBot: (Read) This feature returns one of three health-monitoring temperatures from the camera control console (i.e., controller). The value returned is the temperature of the top of the console processor board. The value is in degrees Celsius. Min: -100, Max: 250

NOTE: For the single-head controller, the bottom and power supply temperature readings share the same sensor.

ConsoleTempPS: (Read) This feature returns one of three health-monitoring temperatures from the camera control console (i.e., controller). The value returned is the temperature of the bottom of the console processor board. The value is in degrees Celsius. Min: -100, Max: 250

ConsoleTempTop: (Read) This feature returns one of three health-monitoring temperatures from the camera control console (i.e., controller). The value returned is the temperature of the power supply. The value is in degrees Celsius. Min: -100, Max: 250

DefectConceal: (R/W) Toggles defect concealment for the active camera head. Off/On

DoubleExposureTPD: (R/W) This feature sets/returns the transfer pulse delay for double exposure trigger mode (Mode 4). The allowable range is 0.1 to 999 µsec in 0.1 µsec intervals. However, due to CCD timing constraints, the minimum value that the CCD will actually implement depends on the camera model. For the minimum integration time for your camera, refer to the appropriate camera specification table in "Camera Component Specifications", page 162. Min: 0.1 µsec, Max: 999 µsec

ExposureTimeMax: (Read) This feature returns the Value relating to the maximum exposure time. This value varies based on changes in Binning, CamPixClkSpeed and SensorTapReadout. Min: 0 msec, Max 999 msec

FlatFieldNormalization: (R/W) This feature controls whether the flat field gain and offset correction for the currently selected head is active or not. When "on" all pixel values for the sensor are corrected with the values specified in the normalization tables. A separate correction table is maintained for each sensor. Checked/Unchecked

GreenGain: (R/W) This feature sets/returns the analog gain value for the specified head (or tap) where appropriate). Head and sensor tap are set by previously defined individual commands. Min: 1, Max: 63

HeadFanState: (R/W) This feature turns the head cooler fan on/off state for a camera with a fan. On/Off

HeadSensorTargetTECTemp: (Read) This feature returns the temperature (in Celsius) of the camera head sensor TEC. Min: -20, Max: 60

HeadTECState: (R/W) This feature turns the TEC on or off for the camera head sensor. The TEC state must be set to Auto if you want to activate a target temperature setting entered via the **HeadSensorTECTargetTemp**. Off/On/Auto

HeadTargetTECTemp: (R/W) This feature sets/returns the target temperature (in Celsius) of the camera head sensor TEC. This target temperature setting will be only used if HeadSensorTECState has set the TEC state to Auto. Min: -20, Max: 60

HeadTemperature: (Read) This feature returns the temperature (in Celsius) of the camera. Used for cameras that do not contain a TEC. For cameras containing a TEC, refer to the TEC-related commands (for example, HeadTECState). Min: -100, Max: 250

HeadsAttached: (Read) This feature queries the camera and returns the number of heads currently attached to the controller. Min: 1, Max: 1

HeadsSupported: (Read) This feature returns the maximum number of heads that this controller can support. Min: 1, Max: 1

MechanicalShutter: (R/W) This feature sets/returns the state of the mechanical shutter. "0" = locked closed; "1" = trigger activated; "2" = locked open. Min: 0, Max: 2

NOTE: Mechanical shutters are present in Princeton Instruments' full frame Camera Heads, the ES 1602/1603 and ES 3200. The mechanical shutter functions have no effect on Camera Heads employing interline sensors.

Mode6Interval: (R/W) When operating in triggered Mode 6 (Periodic Interval), which causes the camera to self-trigger an image repeatedly at a specified time interval (in milliseconds), this feature is used to set/return the time period between triggers. The typical range is 1/Max frame rate of the sensor to 5 minutes. Min: 1, Max: 32767

PixelClockFrequency: Returns the floating point value for the current pixel clock frequency in MHz for the active camera. Min: 0, Max: 40

RedGain: (R/W) This feature sets/returns the red analog gain value for the camera (or tap) where appropriate). Camera and sensor tap are set by other defined individual features. Min: 1, Max: 63

SensorTapReadout: (R/W) This feature sets/returns the mode to be used when reading pixel data from the sensor. Currently supported mode selections include single tap and dual tap. This mode applies to all sensors attached to the camera. If the sensors in the camera heads do not support dual tap operation, the readout will be single tap and this command will have no effect on operation. Not all camera heads will support dual tap readout. The *MegaPlus* full-frame heads including the ES 1602*/3 and ES 3200 do not support dual tap operation. The effect of this function in custom mixed-sensor configurations will be defined per application. One= single tap/Two=dual tap

NOTE: Sensor tap readout configuration does not affect the camera's data output configuration. When a sensor is read in dual tap mode, the two taps are interleaved in the camera back into a single image stream. Not all camera heads will support dual tap readout. The *MegaPlus* full-frame heads including the ES 1602*/3 and ES 3200 do not support dual tap operation.

StrobeDelay: (R/W) This feature sets/returns the delay between the time that the trigger signal is received and the time that the strobe signal is output. Delay is in milliseconds. If a delay of zero is specified, the strobe will be output immediately after receipt of the trigger. If a delay period is specified that is longer than the integration time of the sensor (determined either by programmed value or pulse width), the strobe will be output immediately at the end of the integration period. Min: 0 msec, Max: 999.00 msec

StrobePolarity: (R/W) This feature sets/returns the output strobe signal polarity. Negative/Positive

TriggerInputPolarity: (R/W) This feature sets/returns the trigger polarity setting for the controller. A positive polarity means the trigger signal will be considered active when the signal is in a high or true state. Negative polarity is the reverse - the trigger signal will be considered active when the signal is in a low state. Negative/Positive

TriggerIntTime: (R/W) When operating in triggered modes that use a programmable integration time, this command specifies the integration time that will be used for the triggered exposure. This control offers a range of integration time values that can be beyond the range available with the standard free-run sensor integration time. The specified integration time will be applied to all Camera Heads when triggering is enabled and a trigger is received. **Note:** In some trigger modes (for example, Mode 1) integration time is controlled by other means, such as pulse width. In this case the Triggered Integration Time value has no effect. Unit is milliseconds. Specified as floating point with 3 digits of fractional precision (x.xxx) to provide 1 µsec resolution. Range is 0.001 (1 µsec) to 10,000 msec (10 seconds) for trigger modes 0-6. For

* ES 1602 has been discontinued.

trigger mode 7, range is 0.010 (10 µsec) to 600,000 (10 minutes). Min: 0.01, Max: 600,000

Important Things to Know

The Triggered Integration Time value applies to all attached camera heads. This value will be applied in trigger mode 0 (Edge Trigger, Asynchronous Reset), 6 (Periodic Interval), or 7 (Overlapped Edge Mode) active.

This function controls triggered integration time only. When the camera is operating in free-run (video) mode, integration time is controlled via the ExposureTimeAbs feature.

Triggered mode operation provides a larger range of integration times. Free-run integration time is limited to one frame-readout time at the current clock rate.

TriggerState: (R/W) This feature turns triggering ON or OFF. When triggering is enabled (checked), the camera will recognize and respond to trigger events in the manner dictated by the trigger mode. When triggering is disabled (unchecked), all trigger signals will be ignored and the camera will operate in a free-run video mode. Checked/Unchecked

TriggeringMode: (R/W) This feature sets/returns the trigger mode for the controller. Note: In addition to setting the trigger mode, triggering must be enabled with the **TriggerState** feature in order for triggered operation to take effect. EdgeControlled, LevelControlled, DoubleExposureMode, PeriodicInterval, OverlappedEdge

TriggeringSource: (R/W) This feature specifies/returns the input source that will initiate the trigger event. Sources can include: external trigger (via BNC on controller rear panel) or software trigger. The software trigger command only applies to mode 0 (Edge Trigger, Asynchronous Reset) and 7 (Overlapped Edge Mode). Use the **TriggerSoftware** feature to cause a software trigger event. BNC/Software

WhiteBalanceAuto: (R/W) This feature sets the white balance for the currently selected head. The system uses the default head value of 1. The camera will perform a semi-auto white balance. Execute

WhiteBalanceBlue: (R/W) This feature sets/returns the white balance for the camera. The values specified for blue is relative to green, which is the master color. The value for green is always 1.0. Min: 0, Max: 64

WhiteBalanceRed: (R/W) This feature sets/returns the white balance for the camera. The values specified for red is relative to green, which is the master color. The value for green is always 1.0. Min: 0, Max: 64

15.2.4 Device Information

Device Information provides description of the camera and its sensor. The device specific information shown below for configuration attributes is for a MegaPlus EC11000 Color camera and single-head GigE Vision controller as of 07/07/09. The intent is to provide examples of what the user would see. Actual information will vary based on the camera and controller that is being used.

AppVersion: (Read) Retrieves the firmware version of the application software. In this case, the value is 504 which represents version 5.04#.

NOTE: The firmware version is expressed as an integer and should be divided by 100 to express the major and minor versions of the firmware.

DeviceID: (Read) 1100015

DeviceManufacturerInfo: (Read) Release-1.2 (00080602)

DeviceModelName: (Read) Retrieves the camera model name (i.e., MegaPlus EC11000 Color).

DeviceScanType: (R/W) This feature specifies the scan type of the sensor. Typically, this feature is not writable. But some cameras might allow switching between linescan and areascan. **DeviceScanType** can be either:

- **Areascan**: 2D sensor
- **Linescan**: 1D sensor

DeviceUserID: (R/W) User-programmable ID.

DeviceVendorName: (Read) Retrieves the camera/controller manufacturer name (i.e., Princeton Instruments).

DeviceVersion: (Read) Version 1.6 (02.00.03)

FpgaVersion: (Read) Retrieves the version of the Camera FPGA (i.e., 1268).

NOTE: The version is expressed as an integer and should be divided by 100 to express the major and minor versions of the firmware.

HeadSerialNumber: (Read) Retrieves the camera serial number (i.e., 110358).

SerialVersion: (Read) Retrieves the firmware version (Serial Protocol Version) of the Serial port communications software (i.e., 143).

NOTE: The version is expressed as an integer and should be divided by 100 to express the major and minor versions of the firmware.

15.2.5 Image Size Control

Image Size Control lists all features controlling the size of the transmitted image.

BinningHorizontal: (R/W) This feature represents the number of horizontal photo-sensitive cells that must be combined (added) together. This has the net effect of increasing the intensity (or signal to noise ratio) of the pixel and reducing the horizontal resolution (width) of the image. A value of 1 indicates that no horizontal binning is performed by the camera.

BinningVertical: (R/W) This feature represents the number of vertical photo-sensitive cells that must be combined (added) together. This has the net effect of increasing the intensity (or signal to noise ratio) of the pixel and reducing the vertical resolution (height) of the image. A value of 1 indicates that no vertical binning is performed by the camera.

Height: (R/W) This feature represents the actual image height transmitted by the camera (in pixels). Min: 1, Max: Sensor height

PixelCoding: (Read) This feature indicates the coding of the pixels in the image. Raw gives the data in the native format of the sensor. It is mainly used for Bayer sensor. This value must always be coherent with the **PixelFormat** feature.

PixelColorFilter: (Read) This feature indicates the type of color filter that is applied to the image. This value must always be coherent with the **PixelFormat** feature.

PixelFormat: (R/W) This feature indicates the format of the pixel to use during the acquisition. Values of the enumeration and the pixel formatting correspond to the GigE Vision specification. It contains all the information provided by **PixelCoding, PixelSize**, **PixelColorFilter** but combined in one single value.

PixelSize: (Read) This feature indicates the total size in bits of a pixel of the image. This value must always be coherent with the **PixelFormat** feature.

SensorDigitzationTaps: (R/W) This feature represents the number of digitized samples outputted simultaneously by the camera A/D conversion stage.

SensorHeight: (Read) This feature indicates the effective height of the sensor in pixels. Its value must be greater than 0. For linescan sensor, this value is 1. Min: 0, Max: full height of the sensor.

SensorWidth: (Read) This feature indicates the effective width of the sensor in pixels. Min: 0, Max: full width of the sensor.

TestImageSelector: (R/W) This feature selects the type of test image that is transmitted by the camera. **TestImageSelector** can take any of the following values:

- **Off**: Image is coming from the sensor.
- **IPEnginetestPattern**: Image is filled with a repetitive pattern.

Width: (R/W) This feature represents the actual image width transmitted by the camera (in pixels). Min: 1, Max: Sensor width

16. FireWire Camera Control

16.1 Standard Features Implementation

The following IIDC standard features are implemented in the camera:

Gain: To accommodate gains at a user level specified as n.n dB (1 decimal point precision) multiply this value x10 and pass in the standard feature value field. The value returned from the camera will also be specified as value times 10. This also applies to the maximum and minimum value fields of the properties register.

Integration Time: To accommodate integration time values at a user level specified as n.n msec (1 decimal point precision) multiply the desired integration time x10 and pass in the standard feature value field. The value returned from the camera will also be specified as value x 10. This also applies to the maximum and minimum value fields of the properties register.

16.2 Advanced Features

The IIDC standard supports the concept of vendor-unique features that are not specified by the standard IIDC camera control features. Each vendor must define Control and Status Registers (CSRs) for these additional features and write the base address of these CSRs at offset 480h as a quadlet offset value from the base address of the initial register space.

In the advanced feature address space, the first two quadlets are defined by the IIDC standard as an Access Control Register (ACR). The IIDC standard specifies a format for locking a specific feature by writing the "feature ID" to this access register and specifying a time-out value. *MEGAPLUS* does not utilize the feature locking function.

The base address of the Advanced Features Access Register is: Bus ID, Node ID, FFFF F002 1000

The presence of advanced features is identified by setting Bit 0 of the BASIC_FUNC_INQ register at offset 400h. This base address is stored in the Advanced_Feature_Inq register at offset 480h.

16.3 IEEE 1394a FireWire Image Formatting

IEEE 1394a applies to multi-head controller only.

The *MEGAPLUS* camera conforms to the IIDC Version 1.30 standard for camera control and image output formatting.

The IIDC protocol defines seven video image formats, and within each format, a variety of video modes. All currently available *MEGAPLUS* Camera Heads support Format 7, the partial image size format. Format 7 is used for cameras with resolutions or frame rates that do not match the other video modes, and is additionally used to support a Region of Interest (ROI) capability to read and transmit only a portion of the acquired image data instead of the entire image.

16.3.1 Format 7 Video Modes and Binning

MEGAPLUS cameras support four different Format 7 video modes as listed in the table below. Modes 1, 2, and 3 are used to support binned operation of the sensor.

Format 7 Video Mode	Image Output Format
Mode 0	Full Sensor Resolution
Mode 1	2 x 2 Binning
Mode 2	3 x 3 Binning
Mode 3	4 x 4 Binning

Table 16.1 Video Modes and Image Output Formats

Binning is a special sensor readout mode in which the signal level from adjacent pixels is added together, which has the effect of creating a virtual pixel with a larger area and increased sensitivity. Using an ES 4020 as an example, the maximum image size for Mode 0 is the sensor resolution of 2048 x 2048. With 2x2 binning, the resolution is half of the full size at 1024 x 1024, with 3x3 binning it is 683 x 683, and with 4x4 binning it is 512 x 512.

It is important to be aware that binning level and the 1394 Format 7 video mode are directly linked within the camera. Changing the binning level via the Steepening command will automatically change the camera's video mode and vice versa. Therefore, when writing code to operate the camera, make sure that both values are refreshed any time a change is made to either.

Binning level is Camera Head attribute - i.e., binning level can be set for each individual head. However, when the camera is in the Four-Head Mono configuration, all heads must be the same resolution and changing the binning level on one head will change the binning level to the same value for all heads. In the Two-Head Color configuration, each head can have a different binning level. Update the Format 7 image size values and video mode information whenever switching the camera to a new output head.

16.3.2 Format 7 Color Coding

Each of the *MEGAPLUS* Format 7 video modes can support three different IIDC color codings. *MEGAPLUS* supports the IIDC RGB8, MONO8, and MONO16 formats. These formats are described by the IIDC standard as follows:

Color Code	Description
RGB	RGB = 8 bits, non-compressed
MONO8	Y only. Y = 8 bits, non compressed
MONO16	Y only. Y = 16 bits, non-compressed

The configuration of the IIDC color-coding setting is independent of the camera's bit window setting. The bit depth and sensor readout mode (i.e., RGB color vs. raw Bayer data for color sensors) affect the formatting of the raw pixel data produced by the camera. Bit depth is controlled by the Bit Window command. Readout is controlled by the Mode parameter of the QuickMux command. The IIDC color coding affects the way this raw pixel data is packed into 1394 data packets for transmission to a host. The following table shows how the settings combine:

Color Coding	Bit Depth	Sensor Type	Readout	Results
MONO8	8	Mono	Mono	8-bit mono pixels
MONO8	8	RGB	RGB	Green image plane, 8-bit
MONO8	8	RGB	Raw Bayer	Raw Bayer data, 8-bit
MONO8	10	Mono	Mono	8 LSBs of 10-bit mono pixels
MONO8	10	RGB	RGB	Green image plane, 8 LSBs of 10
MONO8	10	RGB	Raw Bayer	Raw Bayer, 8 LSBs of 10
MONO8	12	Mono	Mono	8 LSBs of 12-bit mono pixels
MONO8	12	RGB	RGB	Green image plane, 8 LSBs of 12
MONO8	12	RGB	Raw Bayer	Raw Bayer data, 8 LSBs of 12
Mono16	8	Either	-	Not Supported
Mono16	10	Mono	Mono	16-bit pixels with data in 10 LSBs
Mono16	10	RGB	RGB	Green image plane, 16-bit pixels with data in 10 LSBs
Mono16	10	RGB	Raw Bayer	Raw Bayer data, 16-bit pixels with data in 10 LSBs
Mono16	12	Mono	Mono	16 bit pixels with data in 12 LSBs
Mono16	12	RGB	RGB	Green image plane, 16-bit pixels with data in 12 LSBs
Mono16	12	RGB	Raw Bayer	Raw Bayer data, 16-bit pixels with data in 12 LSBs
RGB8	8	Mono	Mono	R G B 8-bits mono data
RGB8	8	RGB	RGB	RGB color image, 8-bit pixels
RGB8	8	RGB	Raw Bayer	R G B 8 bits raw Bayer data
RGB8	10	Mono	Mono	R G B 8 LSBs of 10-bit mono data
RGB8	10	RGB	RGB	RGB color image, 8 LSBs of 10-bit data
RGB8	10	RGB	Raw Bayer	R G B 8 LSBs of 10-bit raw Bayer data
RGB8	12	Mono	Mono	R G B 8 LSBs of 12-bit mono data
RGB8	12	RGB	RGB	RGB color image, 8 LSBs of 12-bit data
RGB8	12	RGB	Raw Bayer	R G B 8 LSBs of 12-bit raw Bayer data

Table 16.2 IEE-1394 FireWire Data Packets Settings

16.3.3 Partial Image or ROI Format

The IIDC Format 7 video mode also supports the ability to acquire and transmit only a portion of the full image when desired. This can be useful in eliminating the transmission and processing of unnecessary data and therefore speeding up system throughput. ROI increases communication throughput by eliminating the transmission of unwanted data, but it has no effect on the frame rate of the sensor. The sensor will continue to operate at its native frame rate (unless in triggered mode).

To use the partial image format, define a rectangular sub-region of interest within the full image. This rectangle is defined by specifying the coordinates of the top, left pixel and the width and height of the desired region. The granularity of the ROI, the smallest allowable size and position, is determined by the Unit Width and Unit Height values specified by the camera. These values can be obtained from the IIDC Format 7 UNIT_SIZE_INQ and UNIT_POSITION_INQ registers (or from the corresponding camera attributes in IMAQ for 1394). *MEGAPLUS* currently supports a minimum unit size of 1 column x 2 rows.

The ROI capability is accessible through MEGAPLUS Central by changing the output window coordinates in the FireWire Transfer section of the Data Control Panel.

NOTE: ROI is available only for IEEE 1394a image data output. This feature is a special mode of IEEE 1394a data transmission and has no affect on CameraLink image data output.

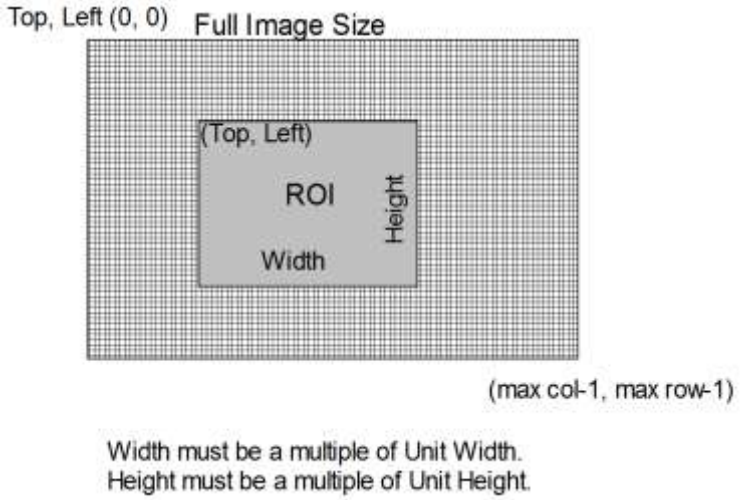


Figure 16.1 Format 7 ROI Capability allows Capture of Partial Images

16.3.4 Access Control Register

The Advanced Feature Access Control Register (ACR) consists of the first two quadlets of the Advanced Features Address Space. The locking mechanism for the ACR specified by IIDC Standard Version 1.3 is not implemented for the *MEGAPLUS* cameras.

In the descriptions below, the offset field is the byte offset from the Advanced Features Base Address.

Offset	Name	Notes
000h	Access_Control_Reg	Reserved , but not utilized

Version Inquiry Register

Returns version information for camera firmware.

Offset	Name	Field	Bit	Description
010h	VERSION_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
			[1..7]	Reserved
		Length (Read only)	[8..15]	Specifies the length in quadlets of the string data in the Version information register at offset 1014h .
			[16..31]	Reserved

Version Information Register

Contains the camera version information indicated by the VERSION_INQ register.

Offset	Name	Field	Bit	Description
014h	VERSION_INFO	Version String	[n Bytes]	An ASCII character string that returns the version numbers for the camera. The length of this string field is equal to the number of quadlets given in the "Length" field of the VERSION_INQ register at offset 1010h.

The string information in the VERSION_INFO string is as follows:

aa.aa bb.bb cc.cc

where :

aa.aa is the camera firmware version

bb.bb is the camera FPGA version

cc.cc is the serial protocol version

If the string does not fill the entire allocated field length, it will be padded with 0x00 at the end of the string.

Max Heads Supported Register

Returns info on maximum heads supported by this controller.

Offset	Name	Field	Bit	Description
040h	MAXHEADS_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		--	[1.7]	Reserved
		Max Heads Supported (Read only)	[8.11]	Indicates the number of Camera Heads this controller can support.
		---	[12..31]	Reserved

Max Heads Attached Register

Reports the number of Camera Heads currently attached to the controller.

Offset	Name	Field	Bit	Description
044h	MAXHEADS_ATTACHED_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		--	[1..7]	Reserved
		Max Heads Attached (Read only)	[8..11]	Indicates the number of Camera Heads currently attached to this camera
		---	[12..31]	Reserved

Head Configuration Registers

The Camera Head configuration information is found in the HEAD_CFG_INQ1 and HEAD_CFG_INQ2 registers at offsets.

Offset	Name	Field	Bit	Description
048h	HEAD_CFG_INQ1	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read/Write)	[1..4]	The head number to report info on. Writing this value causes the configuration info for this head to be loaded into the register.
		No Sensors (Read only)	[5..7]	The number of sensors in this Camera Head.
		No Taps (Read only)	[8..13]	The number of taps per sensor in this Camera Head.
		Spectral Type (Read only)	[14..16]	The spectral type of the Camera Head. 0 = monochrome 1 = Bayer color 2 = RGB 3 = CIR [others reserved]
		Serial No	[17..31]	Serial number of the Camera Head.
04Ch	HEAD_CFG_INQ2	Horizontal Resolution	[0..15]	Indicates the horizontal resolution of the Camera Head in pixels.
		Vertical Resolution	[16..31]	Indicates the vertical resolution of the Camera Head in pixels.

Active Head Register

Determines the Camera Head to be identified as the currently active head. All subsequent commands that affect Camera Head performance will be applied to this Camera Head.

NOTE: Before issuing this command, you must set the ACTIVE_HEAD_INQ register (050h) to the head number you would like this command to apply to. The value returned in the Head No field will contain the head number of the currently active head.

Offset	Name	Field	Bit	Description
050h	ACTIVE_HEAD_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read/Write)	[1..4]	The active head number. Writing this value sets the currently active head. Reading this value reports the currently active head. Default-Head 1.
			[5..31]	Reserved

16.3.5 Configuration Selection Register

Sets a value in the camera that will determine which configuration the camera will boot the next time the camera controller is powered On or Off. The configuration is specified by an index number that is passed as a parameter. This selection flag is written to the camera's non-volatile memory. The next time the camera is turned On this value will be read and used to control the way the camera configures itself.

The *MEGAPLUS* camera can support a variety of different configurations and features depending on the firmware contents that are loaded (FPGA configuration and processor software). This configuration ID flag is used to tell the camera which version of the firmware is loaded the next time the camera is started.

Offset	Name	Field	Bit	Description
054h	CFG_SELECT_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
			[1..7]	Reserved
		Curr_cfg (Read only)	[8..15]	The configuration identified of the currently loaded firmware. 1 = Dual head, adv color 2 = Four head mono
		Cfg_Flag (Read/Write)	[16..23]	Writing this value sets the flag for the configuration to be loaded the next time the camera is rebooted.
			[24-31]	Reserved

Crosshair Display Register

This register controls the display of the crosshair indicator in the image data for the specified Camera Head. The crosshair boundary indicates the region of the image used for semi-auto white balance calculations. The indicator is positioned in the exact digital center of the image and is useful for camera targeting.

Offset	Name	Field	Bit	Description
05Ch	CROSSHAIR_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read/Write Only)	[1..4]	The head the returned information applies to. The head this inquiry applies to Default-Head 1
		--	[5..15]	Reserved
		Crosshair State (Read/Write)	[16]	Sets or reports crosshair display state. 0 = off 1 = on
			[17-31]	Reserved

WARNING: The crosshair display is produced by setting pixel values in the image to display the crosshairs, therefore it supersedes the original image pixel values. disable the display of the crosshair when you are ready to acquire original images.

16.3.6 Mechanical Shutter Register

This register controls the mechanical shutter for a specified Camera Head. This function only applies to those Camera Heads that implement a mechanical shutter. Operation of the shutter can be enabled or disabled. When disabled, the feature specifies whether the shutter is locked in open or closed position.

The "presence" value indicates if the shutter control capability is available in the camera feature set. It is not an indication if this particular head has a shutter present. Therefore, if the feature is available, the is_sup value will be 1, regardless of the camera head type.

Mechanical shutters are present in Princeton Instruments' full frame Camera Heads, the ES 1602*/1603 and ES 3200. The mechanical shutter functions have no effect on Camera Heads employing interline sensors.

Before issuing this command, you must set the ACTIVE_HEAD_INQ register (050h) to the head number you would like this command to apply to. The value returned in the Head No field will contain the head number of the currently active head.

Offset	Name	Field	Bit	Description
060h	MECHSHUTTER_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read/Write only)	[1..4]	The head this inquiry the returned information applies to. Writing causes the camera to read the current shutter state information. Default-Head 1
			[5..15]	[Reserved
		Shutter State (Read/Write)	[16]	Enables or disables shutter operation for the specified Camera Head. 0 = disabled 1 = enabled
		LockPos (Read/Write)	[17]	Specifies what position the shutter should be locked in when disabled. 0 = closed 1 = open
		--	[18..31]	Reserved

16.3.7 Defect Correction Register

This register controls the camera's feature to conceal defective pixels. Defective pixels are identified in a defect list table stored in the camera console. This register controls and reports the status of defect concealment for a specified imaging head.

NOTE: Before issuing this command, you must set the ACTIVE_HEAD_INQ register (105h) to the head number you would like this command to apply to. The value returned in the Head No field will contain the head number of the currently active head.

Offset	Name	Field	Bit	Description
106Ch	DEFECT_STATE_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read only)	[1..4]	The head the returned information applies to.
		---	[5..7]	Reserved
		Defect Conceal State (Read/Write)	[8]	Sets or reads the state of defect concealment for the specified imaging head. 0 = defects not concealed 1 = defects concealed
		---	[9..31]	Reserved

* ES 1602 has been discontinued.

16.3.8 Normalization State Register

This register controls and reports the state of flat field normalization for the specified imaging head.

NOTE: Before issuing this command, you must set the ACTIVE_HEAD_INQ register (105h) to the head number you would like this command to apply to. The value returned in the Head No field will contain the head number of the currently active head.

Offset	Name	Field	Bit	Description
1070h	FF_STATE_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read only)	[1..4]	The head the returned information applies to.
		---	[5..7]	Reserved
		Normalization State (Read/Write)	[8]	Sets or reads the state of flat field normalization for the specified imaging head. 0 = disabled 1 = enabled
		---	[9..31]	Reserved

16.3.9 Pixel Clock Speed Register

The pixel clock feature allows operation of a Camera Head at a "high speed" or "low speed." The high speed setting operates the sensor at its maximum clock speed for maximum throughput. The low speed setting operates the sensor at a reduced rate (nominally 50% of maximum) to achieve slower, but lower noise operation. Use the PIXCLKFREQ_INQ register to determine the actual pixel clock rate.

The feature allows the user to optimize camera operation for the requirements of the application. Clock "speeds" (i.e. high/low) are selected rather than specific clock frequencies, which are sensor-dependent. The "high-speed" setting is the factory default and runs the sensor at its maximum rate. The camera base pixel clock frequency can be determined with the Pixel Clock Speed Inquiry Register. The actual camera head pixel clock speed is dependent on readout configuration and sensor type.

Changing the speed selection causes quite a bit of processing in the camera as all of the timing generators are reset and frequency dependent values are recalculated. When a speed change is initiated, the Busy bit goes high and remains high until the speed change processing is completed.

In multi-head camera configurations where data from multiple heads is output simultaneously via CameraLink, the pixel clock speed for all Camera Heads must be the same. In these configurations, setting the clock speed for one head will affect the clock speed for all heads.

Offset	Name	Field	Bit	Description
064h	PIXCLKSPEED_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read/Write)	[1..4]	The head this inquiry applies to. Writing causes the camera to update the clock speed field with the current setting for this camera head. Default: Head 1
		---	[5..15]	Reserved
		Clock Speed (Read/Write)	[16]	Sets or reads the clock speed setting for the specified camera head. 01=low speed selection 10=high speed selection
			[17..31]	Reserved

16.3.10 Pixel Clock Frequency Register

This register reports the actual pixel clock frequency for the specified Camera Head. The frequency value is returned in MHz. This command reports the camera's core pixel clock base frequency. The pixel clock frequency for the imaging sensor may be an integral divisor of this frequency.

Offset	Name	Field	Bit	Description
068h	PIXCLKFREQ_INQ	Presence (Read only)	[0]	Indicates presence of feature
		Head No (Read/Write)	[1..4]	The head this inquiry applies to. Writing causes the camera to update the clock frequency field with the current setting for this camera head. Default: Head 1
		---	[5..17]	Reserved.
		Clock Frequency (Read only)	[18..31]	Reads the pixel clock frequency for the specified camera head. 0 = low speed selection 1 = high speed selection

16.3.11 Triggers

Trigger Source Register

The trigger source register specifies the source of the trigger signal. The trigger signal can be sourced from the external BNC on the rear panel of the camera, the CameraLink CC1 signal, or via a software trigger.

The software trigger command only applies to mode 0 (Edge Trigger, Asynchronous Reset). Use the SoftwareTrigger() command to cause a software trigger event.

Offset	Name	Field	Bit	Description
074h	TRIG_SOURCE_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		---	[1..7]	Reserved
		Trigger Source (Read/Write)	[8..9]	Specifies the source for the trigger signal. 0 = external BNC 1 = CameraLink CC1 2 = Software
		--	[10..31]	Reserved

Trigger Integration Time

When operating in triggered modes that use a programmable integration time, this command specifies the integration time that will be used for the triggered exposure. This control offers a range of integration time values that can be beyond the range available with the standard free-run sensor integration time. The specified integration time will be applied to all Camera Heads when triggering is enabled and a trigger is received. In some trigger modes (for example, Mode 1) integration time is controlled by other means, such as pulse width. In this case the Triggered Integration Time value has no effect.

Important Things to Know

"The Triggered Integration Time value applies to all attached camera heads. This value will be applied in trigger mode 0 (Edge Trigger, Asynchronous Reset), or 6 (Periodic Interval).

"This function controls triggered integration time only. When the camera is operating in free-run (video) mode, integration time is controlled via the SetIntTime() command.

Triggered mode operation provides a larger range of integration times. Free-run integration time is limited to one frame-readout time at the current clock rate.

The trigger integration time register specifies the integration time value for trigger modes that use a programmable integration time value. The integration time value is specified in milliseconds.

Offset	Name	Field	Bit	Description
078h	TRIG_INTTIME_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		--	[1..7]	Reserved
		Integration Time (Read/Write)	[8..31]	Specifies the integration time value in microseconds. Write to set the value. Read to report the current value. Min 10 µs, max 10 million µs (10 seconds).

Mode 6 Interval

When operating in triggered mode 6, which causes the camera to self-trigger an image repeatedly at a specified time interval, this register is used to control the time period between triggers. The value is an integer number of milliseconds. The typical range is 1/Max frame rate of the sensor to 5 minutes 32 seconds.

Offset	Name	Field	Bit	Description
0Ach	MODE6_INT_INQ1	Presence (Read only)	[0]	Indicates presence of this feature.
		--	[1..7]	Reserved
		Min Value (Read only)	[8..15]	Specifies the minimum interval time in milliseconds. (Typical = 1)
		Max Value (Read only)	[16..31]	Specifies the maximum interval time in milliseconds. (Typical = 32,767)
0B0h	MODE6_INT_INQ2	Interval Value (Read/Write)	[0..15]	Indicates Mode 6 frame interval in milliseconds.
		---	[16..31]	Reserved

Double Exposure TPD Value

This register controls the transfer pulse delay for double exposure trigger mode (Mode 4). Typical range is 1-10,000 microseconds (10msec) in 0.1 µs intervals. Register parameter values are unsigned integers and do not support floating point values. Therefore, the parameter value is the number of 0.1 µs units i.e. the value in the register is TPD * 10.

Offset	Name	Field	Bit	Description
0B4h	TPD_INQ1	Presence (Read only)	[0]	Indicates presence of this feature.
		---	[1..3]	Reserved
		Min Value (Read only)	[4..11]	Specifies the minimum TPD interval time in tenths of microseconds.
		Max Value (Read only)	[12..31]	Specifies the maximum TPD time in tenths of microseconds.
0B8h	TPD_INQ2	Interval Value (Read only)	[0..19]	TPD Value in tenths of microseconds
		--	[20..31]	Reserved

Software Trigger

This function causes the controller to issue a software trigger event which will cause the controller to respond in the same manner as if it has received an external trigger signal. When the camera receives this command, it will cause a trigger event for the current trigger mode. For the event to be acknowledged by the camera, the following conditions must be met. The camera should be configured for a trigger mode compatible with software triggering, the trigger source should be set to software, and triggering should be enabled.

The software trigger command only applies to mode 0 (Edge Trigger, Asynchronous Reset) Note: The camera trigger state must be enabled for this command to have any effect.

Offset	Name	Field	Bit	Description
0C0h	SOFTWARE_TRIGGER	Presence (Read only)	[0]	Indicates presence of this feature.
		---	[1..7]	Reserved
		Trigger Event (Write only)	[8]	Write a value of 1 to cause a trigger event. The controller will clear this bit after it is received. Read has no effect.
		---	[9..31]	Reserved

16.3.12 Strobe Inquiry Register

The strobe inquiry register provides access to the strobe output feature. Register fields control strobe signal polarity and the delay from the beginning of sensor integration time to the output of the strobe signal. If a delay of zero is specified, the strobe will be output immediately after receipt of the trigger. If a delay period is specified that is longer than the integration time of the sensor (determined either by a programmed value or pulse width), the strobe will be output immediately at the end of the integration period.

Offset	Name	Field	Bit	Description
07Ch	STROBE_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		--	[1..7]	Reserved
		Polarity (Read/Write)	[8]	Write to change the polarity of the strobe. Read to report the current value. 0 = Low active output. 1 = High active output.
			[9..11]	Reserved
		Delay (Read/Write)	[12..31]	Strobe delay in microseconds. Write to change the delay. Read to report the current value. Mix = 0 Max = 1,000,000 µs (1 second)

16.3.13 Bit Window Register

The Bit Window register specifies the system pixel data bit depth and the LSB of the bit window. The LSB value can be used to adjust which of the available data bits are included in the output.

Offset	Name	Field	Bit	Description
080h	BITWINDOW_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		---	[1]	Reserved
		Min Value (Read only)	[2..8]	Specifies the minimum bit depth that can be specified
		Max Value (Read only)	[9..15]	Specifies the maximum bit depth that can be specified.
		Bit Depth (Read/Write)	[16..20]	Write to set bit depth. Read to report current value. Bit depth must be increments of 2 starting at 8 bits. (i.e. 8, 10, or 12)
		LSB (Read/Write)	[21..25]	Write to specify LSB. Read to report current setting. Specifies bit to use as LSB of output. Possible values depend on bit depth setting. 12 bit data, LSB = 0 10 bit data, LSB = 0,1,2 8 bit data, LSB = 0,1,2,3,4

16.3.14 Temperature Registers

Controller Temperature Register

The controller temperature register reports current temperature of the camera control controller. The values of three different temperature monitors are reported.

Offset	Name	Field	Bit	Description
088h	CONTROLLER_TEMP_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		---	[1-7]	Reserved
		Controller Top Temperature (Read only)	[8..15]	Reports temperature on top of controller processor board in degrees Celsius.
		Controller Bottom Temperature (Read only)	[16..23]	Reports temperature on bottom of controller processor board in degrees Celsius
		Controller Power Supply Temperature (Read only)	[24..31]	Reports temperature of controller power supply board in degrees Celsius.

Head Temperature Register

Reports ambient head temperature. The temperature sensor is not in close proximity to the CCD.

Offset	Name	Field	Bit	Description
58h	HEAD_TEMP_REG	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read/Write)	[1..4]	The head this inquiry applies to. Writing causes the camera to read the current temperature. Default: Head 1
		--	[5..15]	Reserved
		Head Temp (Read only)	[16..23]	The imaging head temperature. This is a signed (twos-complement) value containing the number of whole degrees Celsius.
		Head Temp, 1/100ths (Read only)	[24..31]	The imaging head temperature. This is an unsigned count of the number of 1/100th degrees Celsius.

Imaging Head TEC Temperature Register

Reports the temperature of the specified imaging head's CCD in degrees Celsius and in hundredths of degrees Celsius. For EC models only. ES models will read 0.

Offset	Name	Field	Bit	Description
10e8h	HEAD_TEMP_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read/Write)	[1..4]	The head this inquiry applies to. Writing causes the camera to read the current temperature. Default: Head 1
		--	[5..15]	Reserved
		Head Temp (Read only)	[16..23]	The imaging head TEC temperature. This is a signed (twos-complement) value containing the number of whole degrees Celsius.
		Head Temp, 1/100ths (Read only)	[24..31]	The imaging head TEC temperature. This is an unsigned count of the number of 1/100th degrees Celsius.

16.3.15 Controller Reset Register

This function causes the controller to re-initialize itself. The initialization process is a "soft boot" process similar to that performed when the controller power is cycled

Offset	Name	Field	Bit	Description
08Ch	CONTROLLER_RESET_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		---	[1..7]	Reserved
		Controller Reset (Write only)	[8]	Write a value of 1 to cause a reset of the controller hardware. Read has no effect.
		---	[9...31]	Reserved

Quick Mux Register

This register configures the camera data paths to output image data from a specific Camera Head or from a sequence of Camera Heads. The format of the resulting pixel data stream is a function of the current system bit window setting and the head spectral type (mono/color). The Mode field controls whether the data from a color sensor is output in raw or processed form.

Offset	Name	Field	Bit	Description
084h	QUICKMUX_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Output Sel (Read/Write)	[1..7]	Write this value to set output selection. Read to report current value. 1,2,3,4 = Head no. 99 = Custom output (CameraLink only)
		Mode (Read/Write)	[8]	Specifies format of output data. Write to set. Read to report current value. For monochrome sensors: 0 = mono output For Bayer color sensors: 0 = RGB output 1 = raw Bayer data
		---	[21..31]	Reserved

Controller Reset Register

This function causes the controller to re initialize itself. The initialization process is a "soft boot" process similar to that performed when the camera controller power is cycled

Offset	Name	Field	Bit	Description
08Ch	CONTROLLER_RESET_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
			[1..7]	Reserved
		Controller Reset (Write only)	[8]	Write a value of 1 to cause a reset of the controller hardware. Read has no effect.
		---	[9..31]	Reserved

16.3.16 Color Transform Coefficients Registers

The Color Transform Coefficients registers specifies a set of 9 coefficients for the 3x3 matrix in the color transform engine provided in the advanced color processing configuration of the camera control controller firmware. Coefficients values are passed as 1000 x the floating point coefficient values (i.e., decimal point moved 3 places right) so that coefficients can be passed as integer values.

Offset	Name	Field	Bit	Description
090h	COLOR_COEFF_INQ1	Presence (Read only)	[0]	Indicates presence of this feature.
			[1..7]	Reserved
		Enable (Read/Write)	[8]	Enable/Disable processing of color space registers. 0 = off 1 = on
		--	[9..15]	Reserved
		Coeff (0,0) (Read/Write)	[16..31]	Coefficient for Row 0, Col 0. Write to set. Read to query.
094h	COLOR_COEFF_INQ2	Coeff (0,1) (Read/Write)	[0..15]	Coefficient for Row 0, Col 1. Write to set. Read to query.
		Coeff (0,2) (Read/Write)	[16..31]	Coefficient for Row 0, Col 2. Write to set. Read to query.
098h	COLOR_COEFF_INQ3	Coeff (1,0) (Read/Write)	[0..15]	Coefficient for Row 1, Col 0. Write to set. Read to query.
		Coeff (1,1) (Read/Write)	[16..31]	Coefficient for Row 1, Col 1. Write to set. Read to query.
09Ch	COLOR_COEFF_INQ4	Coeff (1,2) (Read/Write)	[0..15]	Coefficient for Row 1, Col 2. Write to set. Read to query.
		Coeff (2,0) (Read/Write)	[16..31]	Coefficient for Row 2, Col 0. Write to set. Read to query.
0A0h	COLOR_COEFF_INQ5	Coeff (2,1) (Read/Write)	[0..15]	Coefficient for Row 2, Col 1. Write to set. Read to query.
		Coeff (2,2) (Read/Write)	[16..31]	Coefficient for Row 2, Col 2. Write to set. Read to query.

NOTE: Before issuing this command, set the ACTIVE_HEAD_INQ register (050h) to the head number you would like this command to apply to. The value returned in the Head No field will contain the head number of the currently active head.

16.3.17 Color LUT Processing

This register is used to enable and disable application of the three input lookup tables (LUTs) and the three output lookup tables (LUTs) that are a part of the color space conversion/ correction engine. When the color space LUTs are enabled, the pixel values from the sensor are translated by the LUTs. When the color space LUTs are disabled, the pixel values from the sensor are not affected by the LUTs.

Offset	Name	Field	Bit	Description
0A4h	COLOR_LUT_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		---	[1..7]	Reserved
		Enable (Read/Write)	[8]	Enable/Disable processing of color space LUTs. 0 = off 1 = on
		---	[9..31]	Reserved

16.3.18 Sensor Tap Readout

This register controls how the pixel data is read from the sensor. Currently supported mode selections include single tap and dual tap. This mode applies to all sensors attached to the camera. If the sensors in the Camera Heads do not support dual tap operation, the readout will be single tap and this command will have no effect on operation. The effect of this function in custom mixed-sensor configurations will be defined per application.

NOTE: All camera heads will support dual tap readout. The MEGAPLUS full-frame heads including the ES 1602/3 and ES 3200 do not support dual tap operation.

Offset	Name	Field	Bit	Description
0A8h	TAP_READOUT_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
			[1..7]	Reserved
		Tap Configuration (Read/Write)	[8..15]	Select tap readout configuration. 1 = single tap 2 = dual tap
		--	[16..31]	Reserved

16.3.19 1394 Bus Reset Register

This function causes the controller to issue a reset signal on the 1394 bus.

Offset	Name	Field	Bit	Description
0BCh	1394_RESET_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		--	[1..7]	Reserved
		1394 Bus Reset (Write only)	[8]	Write a value of 1 to cause the controller to issue a 1394 bus reset. Read has no effect.
		--	[9..31]	Reserved

16.3.20 Binning

This register controls binning functions in the camera. Binning is a technique whereby the signals from adjacent pixels in a CCD are combined to produce an effective array with larger pixels, lower resolution, and faster frame rates. Binning is supported for monochrome sensors only. The *MEGAPLUS* software and firmware allow binning levels to be specified for a color camera head and the camera will apply binning for that head. However, it should be noted that the binning process is not color-aware and will not account for pixel color in the binning process. As a result, the colors in the binned image will be incorrect.

Before issuing this command, you must set the ACTIVE_HEAD_INQ register (050h) to the head number you would like this command to apply to. The value returned in the Head No field will contain the head number of the currently active head.

Offset	Name	Field	Bit	Description
0C4h	BINNING_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		Head No (Read only/Write)	[1..4]	The head this inquiry returned information applies to. Writing causes the camera to update the binning level with the current setting for this Camera Head. Default: Head 1. In Four-Head Mono FPGA configuration, binning level is applied to all attached heads.
		--	[5..7]	Reserved
		Binning Level (Read/Write)	[8..15]	Sets or reads the binning level for the specified Camera Head. 1 = 1x1 (no binning) 2 = 2x2 binning 3 = 3x3 binning 4 = 4x4 binning
		Min Level	[16..23]	Min bin level
		Max Level	[24..31]	Max bin level

NOTE: In the MEGAPLUS Four-head monochrome FPGA configuration, the current binning state applies to all heads. Setting the binning level for any head will apply the same binning state to all heads. In the Two-head Advanced Color configuration, each head maintains its own binning state.

16.3.21 Video Control Register

This register controls and reports the state the video output feature of the camera.

Offset	Name	Field	Bit	Description
0D0h	VIDEO_STATE_INQ	Presence (Read only)	[0]	Indicates presence of this feature.
		--	[5..7]	Reserved
		Video Output State (Read/Write)	[8]	Sets or reads the state of the video output feature. 0 = disabled 1 = enabled
		---	[16..31]	Reserved

16.3.22 CCD Analog Color Gain - Red

This register is used to control the analog color gain for the red pixels of a color sensor. The gain count value is a digital number that controls the gain applied by the "'analog front end" signal processing circuitry in the camera head before the signal is digitized. The actual gain applied for a specific value is camera head specific. The table below maps the gain count values to the resulting gains for each camera head.

Camera Head	Min Gain Count	Min Gain Value	Max Gain Count	Max Gain Value
ES 11000	0	-2 dB	63	10 dB
ES 4010/4020	0	0 dB	63	18 dB
ES 2001*/2020/2093	0	0 dB	63	18 dB
ES 1602*/1603	0	-2 dB	63	10 dB
ES 3200	0	-2 dB	63	10 dB

*ES 1602 and ES 2001 have been discontinued.

Table 16.3 Gain Count Values-Red

When using this register, the Updt_Mode bit is used to control when the current value field is updated. Updating a specific color gain is a three step process.

1. Specify the head number to apply this command to by setting the ACTIVE_HEAD_INQ register (050h).
2. Specify sensor and tap with Updt_Mode=0 to select the appropriate color gain value. Reading this register when Updt_Mode=0 will return the appropriate maximum, minimum, and current gain counts for the specified head, sensor, and tap.
3. Specify sensor and tap and the new gain count value with Updt_Mode=1 to update the gain count for the specified head, sensor, and tap.

Offset	Name	Field	Bit	Description
0D4h	CCD_REDGAIN_INQ	Presence (Read only)	[0]	Indicates presence of this feature
		Head No (Read only)	[1..4]	The head containing the returned information
		Sensor	[5..7]	The camera head sensor number with the information. For most heads this value is 1.
		Tap	[8..10]	Sensor tap with this information. a value of 0 applies the gain to all taps In two-tap sensors, 1 = left tap, 2 = right tap.
		Updt_Mode	[11]	Determines if read or write values apply. Value of 0 = return current gain count for specified head, sensor, and tap. Value of 1 = set gain count to current value for specified head, sensor, and tap.
		Min Value (Read only)	[12..15]	Specifies the minimum color gain count value
		Max Value (Read only)	[16..23]	Specifies the maximum color gain count value
		Current Value	[24..31]	Specifies the current red gain count value

16.3.23 CCD Analog Color Gain - Green

This register is used to control the analog color gain for the green pixels of a color sensor. The gain count value is a digital number that controls the gain applied by the "'analog front end" signal processing circuitry in the camera head before the signal is digitized. The actual gain applied for a specific value is camera head specific. The table below specified maps the gain count values to the resulting gains for each camera head.

Camera Head	Min Gain Count	Min Gain Value	Max Gain Count	Max Gain Value
ES 11000	0	-2 dB	63	10 dB
ES 4010/4020	0	0 dB	63	18 dB
ES 2001*/2020/2093	0	0 dB	63	18 dB
ES 1602*/1603	0	-2 dB	63	10 dB
ES 3200	0	-2 dB	63	10 dB

*ES 1602 and ES 2001 have been discontinued.

NOTE: Before issuing this command, set the ACTIVE_HEAD_INQ register (050h) to the head number you would like this command to apply to. The value returned in the Head No field will contain the head number of the currently active head.

When using this register, the Updt_Mode bit is used to control when the current value field is updated. Updating a specific color gain is a three step process.

1. Specify the head number to apply this command to by setting the ACTIVE_HEAD_INQ register (050h).
2. Specify sensor and tap with Updt_Mode=0 to select the appropriate color gain value. Reading this register when Updt_Mode=0 will return the appropriate maximum, minimum, and current gain counts for the specified head, sensor, and tap.
3. Specify sensor and tap and the new gain count value with Updt_Mode=1 to update the gain count for the specified head, sensor, and tap.

Offset	Name	Field	Bit	Description
0D8h	CCD_GREENGAIN_INQ	Presence (Read only)	[0]	Indicates presence of this feature
		Head No (Read only)	[1..4]	The head containing the returned information
		Sensor	[5..7]	The camera head sensor number with the information. For most heads this value is 1.
		Tap	[8..10]	Sensor tap number with this information. A value of 0 applies the gain to all taps. In two-tap sensors, 1 = left tap, 2 = right tap.
		Updt_Mode	[11]	Determines if read or write values. Value of 0 = return current gain count for specified head, sensor and tap. Value of 1= set gain count to current value for specified head, sensor and tap.
		Min Value (Read only)	[12..15]	Specifies the minimum color gain count value
		Max Value (Read only)	[16..23]	Specifies the maximum color gain count value
		Current Value	[24..31]	Specifies the current green gain count value

16.3.24 CCD Analog Color Gain - Blue

This register is used to control the analog color gain for the blue pixels of a color sensor. The gain count value is a digital number that controls the gain applied by the "'analog front end" signal processing circuitry in the camera head before the signal is digitized. The actual gain applied for a specific value is camera head specific. The table below specified maps the gain count values to the resulting gains for each camera head.

Camera Head	Min Gain Count	Min Gain Value	Max Gain Count	Max Gain Value
ES 11000	0	-2 dB	63	10 dB
ES 4010/4020	0	0 dB	63	18 dB
ES 2001*/2020/2093	0	0 dB	63	18 dB
ES 1602*/1603	0	-2 dB	63	10 dB
ES 3200	0	-2 dB	63	10 dB

*ES 1602 and ES 2001 have been discontinued.

When using this register, the Updt_Mode bit is used to control when the current value field is updated. Updating a specific color gain is a three step process.

1. Specify the head number to apply this command to by setting the ACTIVE_HEAD_INQ register (050h).
2. Specify sensor and tap with Updt_Mode=0 to select the appropriate color gain value. Reading this register when Updt_Mode=0 will return the appropriate max, min., and current gain counts for the specified head, sensor, and tap.
3. Specify sensor and tap and the new gain count value with Updt_Mode=1 to update the gain count for the specified head, sensor, and tap.

Offset	Name	Field	Bit	Description
0DCh	CCD_BLUEGAIN_INQ	Presence (Read only)	[0]	Indicates presence of this feature
		Head No (Read only)	[1..4]	The head containing the returned information
		Sensor	[5..7]	The camera head sensor number with the information. For most heads this value is 1.
		Tap	[8..10]	Sensor tap number with this information. A value of 0 applies the gain to all taps. In two-tap sensors, 1 = left tap, 2 = right tap.
		Updt_Mode	[11]	Determines if read or write values. Value of 0 = return current gain count for specified head, sensor and tap. Value of 1= set gain count to current value for specified head, sensor and tap.
		Min Value (Read only)	[12..15]	Specifies the minimum color gain count value
		Max Value (Read only)	[16..23]	Specifies the maximum color gain count value
		Current Value	[24..31]	Specifies the current blue gain count value

17. Connectors, Pin Outs & Cables

17.1 CameraLink MDR Connector

The CameraLink serialized frame grabber interface is compliant with the industry standard CameraLink Specification. This specification is available on the Automated Imaging Association website (**www.machinevisiononline.org**).

Pin Number	Signal Name	Function
1	Inner shield	DC Ground
14	Inner shield	
2	X0-	Data from CameraLink Transmitter
15	X0+	
3	X1-	
16	X1+	
4	X2-	
17	X2+	
5	Xclk-	Transmit Clock from CameraLink Transmitter
18	Xclk+	
6	X3-	Data from CameraLink Transmitter
19	X3+	
7	SerTC+	Serial Communication Data Receive
20	SerTC-	
8	SerTFG-	Serial Communication Data Transmit
21	SerTFG+	
9	CC1-	Connected
22	CC1+	
10	CC2+	N/C
23	CC2-	
11	CC3-	
24	CC3+	
12	CC4+	
25	CC4-	
13	Inner shield	DC Ground
26		

Table 17.1 CameraLink 26 Pin MDR Connector Pin Assignments

17.2 Power Connector Assignment

The Camera Controller has a Lemo elbow receptacle, EPG.0B.302.HLN for 12 VDC power input. This receptacle mates with the FGG.0B.302.CLAD56 plug. Pin 1 is 12VDC and Pin 2 is connected to ground. For more information and availability on the Lemo plug, contact **www.lemousa.com.**

17.3 RS232 Serial DB9 Connector Assignment

Pin Number	Signal Name	Function
1	Tied to 6 and 4	
6	Tied to 1 and 4	
2	TXD	
7	CTS	
3	RXD	
8	RTS	
4	Tied to 1 and 6	
9	N/C	Not Connected
5	Gnd	Ground

Table 17.2 RS232 Serial DB9 Connector Assignment

17.4 Ethernet Cross-over Cable Pin Outs

RJ45 Pin # (END 1)	Wire Color	Diagram End #1	RJ45 Pin # (END 2)	Wire Color	Diagram End #2
1	White/Orange		1	White/Green	
2	Orange		2	Green	
3	White/Green		3	White/Orange	
4	Blue		4	White/Brown	
5	White/Blue		5	Brown	
6	Green		6	Orange	
7	White/Brown		7	Blue	
8	Brown		8	White/Blue	

Table 17.3 Ethernet Cross-over Pin Outs

18. Mechanical

Where appropriate, camera equipment model names have been combined when information applies to several different models. For example, the EC 16000 and EP 16000 may be listed together as EC/EP 16000.

For more information, please contact Princeton Instruments (**www.princetoninstruments.com**).

18.1 Lens Mounts

The type of lens mount on *MEGAPLUS* Camera Heads is dependent upon the image sensor type and size.

Camera Heads	Lens Mounts
EC/EP 16000 EC/EM/EP/ES 11000	Nikon bayonet F-mount or Leica Mount
ES 4020/3200/2093/2020/2001*/1603/1602*	C-mount or F-mount

*ES 1602 and ES 2001 have been discontinued.

Table 18.1 Lens Mounts

18.2 LED Displays

LED (Light Emitting Diode) displays are used on the following models to indicate the state of the camera head equipment.

18.2.1 EC 16000/11000

LED	Lit LED Indication
CCU PWR	Camera head is receiving power from CCU (Camera Controller Unit)
TEC ON	TEC (Thermo Electric Cooler) is cooling the CCD (Charge Coupled Device) sensor
FAN ON	Lit LED indicates the fan (optional) is cooling the CCD sensor

Table 18.2 EC 16000/11000 LED Display

18.2.2 EP 16000/11000

LED	Meaning
CCU PWR	Lit LED indicates the camera head is receiving power from CCU (Camera Controller Unit)
(not used)	(feature not implemented)
FAN ON	Lit LED indicates the fan (optional) is cooling the CCD sensor

Table 18.3 EP 16000/11000 LED Display

18.2.3 EM 11000

LED	Meaning
CCU PWR	Lit LED indicates the camera head is receiving power from CCU (Camera Controller Unit)
(not used)	(feature not implemented)
(not used)	(feature not implemented)

Table 18.4 EM 11000 LED Display

18.3 ES 3200/1603/1602* Mechanical Shuttering

The ES 3200/1603/1602* Camera Heads contain mechanical shutters to provide exposure control with full frame sensors. The shutter open duration is controlled using the integration control for the Camera Head. Generally, mechanical shutters are controlled using the external trigger and operating the camera in triggered mode 0.

For related information see "How to Set the Trigger Parameters" on page 59.

18.4 Camera Component Specifications

Please refer to **www.princetoninstruments.com** for the latest specifications.

18.4.1 EC/EP 16000 Camera Head Specification

Category	Specification
Imaging Device	KAI-16000, CCD, Color and Mono
Sensor Readout	Interline, progressive scan
Pixel Size	7.4 µm x 7.4 µm
Resolution	4872 x 3248, 16 mega-pixels
Imager Size (diagonal)	43.3 mm
Active Area	36 mm (h) x 24 mm (v)
Aspect Ratio	3:2
Saturation Signal	30,000 electrons
Dynamic Range	60 dB
Readout Noise	30 electrons
Shuttering	Electronic
Output Bit Depth	8, 10, or 12 bits per pixel
Camera (Head) Output Ports	Single port to controller
Controller Interface to Head	Digital, proprietary
CameraLink Output Ports	2 - base, medium, dual base
Max. Frame Rate (CameraLink)	3.2 fps @ 30 MHz
Max. Frame Rate (IEEE1394a)	1.6 fps (monochrome)
Sensor Clock Rate	30 MHz (optional over clocked 38 MHz)
Controller Clock Rate	60 MHz (over clocked 76 MHz)
Max. Integration Time	10 Minutes
Anti-blooming	Yes
Max. Frame Rate w/ 2x2 Binning	6.8 fps @ 30 MHz, 6.8 fps @38 MHz
Max. Frame Rate w/ 3x3 Binning	7.5 fps @ 30 MHz, 9.6 fps@ 38 MHz
Max. Frame Rate w/ 4x4 Binning	9 fps @ 30 MHz, 11.7 fps@ 38 MHz
Gain Settings	0-36 dB
Binning	Yes
Cooling Method	EC 16000: Peltier Cooling EP 16000: Passive Cooling
Standard Cooling	EC 16000: ~5 ºC Below Ambient (0 - 40 ºC) EP 16000: ~12 ºC Above Ambient (0 - 40 ºC)
Cooling With Fan Option	EC 16000: ~12 ºC Below Ambient (0 - 40 ºC)
Dark Current Doubling Temperature	7 ºC Incremental
Lens Mount	F-mount
Camera Head Cable	2m, 5m, 7m
Camera Head Dimensions with F-mount (L x D x H)	EC 16000: 123.83 x 89.34 x 98.43 mm EP 16000: 123.83 x 78.01 x 98.43 mm

* ES 1602 and ES 2001 have been discontinued.

Category	Specification
Camera Head Weight with F-mount	EC 16000: 1004 g EP 16000: 645 g
Maximum Head Cable Length	7 meters
Operating Temperature	0 - 40 ºC, non-condensing
Vibration	3 G, sinusoidal from 10 to 500 Hz
Shock	5 G
Power Consumption (Single Head with Controller)	15 W
Certification	FCC Part 15, Class A and CE
CameraLink IEEE-1394a	Base, Medium, Dual Base
DVI	Monitor Output
Serial	RS232
Trigger In	BNC, TTL
Trigger Out	BNC, TTL
Ethernet	10/100 Base-T
FireWire	1394a

Table 18.5 EC/EP 16000 Camera Head Specifications

18.4.2 EC/EM/EP/ES 11000 Camera Head Specification

Category	Specification
Sensor Model	KAI-11002CM with microlens, solid state
Sensor Type	Interline Transfer Progressive Scan CCD
Pixels	4008 (H) x 2672 (V) 10.71 Megapixels
Pixel Size	9.0 µm x 9.0 µm
Photosensitive Area	36.0 mm (H) x 24.0 mm (V)
Imager Size	42.52 mm (diagonal)
Spectral	RGB Bayer Color or Monochrome
Saturation Signal	60,000 e
Bit Depth	8, 10, or 12 bits per color
Peak QE RGB	42%, 37%, 34%
Well Capacity	60,000
Output Bit Depth	Up to 12-bit per channel
Dynamic Range	66 dB full resolution
Gamma	User selectable number
Camera Head Output Ports	Single port to controller
Controller Interface to Head	Proprietary
Camera Output	Single / Dual Tap
CameraLink Output Ports	2
Sensor Output Sensitivity	13µV/e
Shuttering	Electronic
Frame Rate	CameraLink: 4.63 @ 30 MHz FireWire: 3 FPS monochrome
Gain	0-36 dB
Maximum Frame Rate	4.36
Binning Frame Rate	2 x 2: 8.17 FPS @ 30 MHz, or 10.34 FPS @ 38 MHz* 3 x 3: 10.98 FPS @ 30 MHz, or 13.9 FPS @ 38 MHz* 4 x 4: 13.23 FPS @ 30 MHz, or 16.75 FPS @ 38 MHz*
Sensor Clock	30 MHz per channel (can be over clocked to 38 MHz per channel)*
Pixel Clock Rate	30 MHz per channel
Max Data Rate	60 MHz (can be overclocked to 76 MHz)
Lens Adapters	C-mount or F-Mount
Min. integration time in continuous mode	192 µS
Min. integration time in trigger mode	140 µS
Max integration time	>5 sec.
Cooling Method	EC 11000: Peltier Cooling EP 11000: Passive Cooling
Standard Cooling	EC 11000: ~5º C Below Ambient (0-40º C) EM 11000:~12º C Above Ambient (0-40º C)
Cooling with Fan Option	EC 16000: ~15º C Below Ambient (0-40º C)
Dark Current Doubling Temperature	7º C Incremental
Camera Head Dimensions with F-mount (L x D x H)	EC 11000: 123.83 x 89.34 x 98.43 mm EM 11000: 53.98 x 82.24 x 53.98 mm EP 11000: 123.83 x 78.01 x 98.43 mm ES 11000: 113 x 60.8 x 83.2 mm
Camera Head Weight with F-mount	EC 11000: 920 g EM 11000: 284 g EP 11000: 650 g ES 11000: 450 g
Tether Cable Length	2 m, 5 m, 7 m
Operating Temperature	0-40º C non-condensing

Category	Specification
Estimated power consumption (Single Head with Camera Controller)	15 Watts
Estimated power consumption for cooler (EC/EP 11000 only)	5 Watts
Vibration	10-500 Hz, 3G, sinusoidal
Shock	5G

* Running the EC/EM/EP/ES 11000 in an over clocked state (38 MHz) is not recommended.

Table 18.6 EC/EM/EP/ES 11000 Camera Head Specifications

18.4.3 ES 4020 Camera Head Specification

Category	Specification
Sensor Model	KAI-4021CM with microlens, solid-state
Sensor Type	Interline Transfer Progressive Scan CCD
Pixels	2048 (H) x 2048 (V) 4.19 Megapixels
Pixel Size	7.4 µm x 7.4 µm
Photosensitive Area	15.2 mm (H) x 15.2 mm (V)
Imager Size	21.5 mm (diagonal)
Spectral	Bayer RGB Color or Monochrome
Bit Depth	8, 10, or 12 bits per color
Dynamic Range	60 dB full resolution
Saturation Signal	40,000 e
Peak QE RGB	45%, 42%, 35%
Frame Rate	CameraLink: 15 FPS; FireWire: 7.5 FPS monochrome
Gain	0-36 dB
Binning Frame Rates	2 x 2 @ 27.4 FPS; 3 x 3 @ 37.82 FPS; 4 x 4 @ 46.6 FPS
Pixel Clock Rate	Up to 38 MHz per channel
Sensor Output Sensitivity	ES 4020: 31µV/e
Maximum Frame Rate	15
Lens Adapters	C-mount or F-Mount
Min. integration time in continuous mode	104 µS
Min. integration time in trigger mode	75 µS
Max integration time	>5 sec.
Camera Head Dimensions (L x D x H)	54.62 x 82.8 x 54.62 mm with F-Mount adapter 54.62 x 48.5 x 54.62 mm with C-Mount adapter
Camera Head Weight	258 g
Tether Cable Length	2 m, 5 m, 7 m
Operating Temperature	0-40º C non-condensing
Vibration	10-500 Hz, 3G, sinusoidal
Shock	5G

Table 18.7 ES 4020 Camera Head Specifications

18.4.4 ES 3200 Camera Head Specification

Category	Specification
Sensor Model	KAF-3200 ME,
Sensor Type	Full Frame, Progressive Scan CCD
Pixels	2184 H x 1472 V (3.3 Megapixels)
Pixel Size	6.8 µm x 6.8 µm
Photosensitive Area	14.8 mm (H) x 10 mm (V)
Imager Size	16.6 mm (diagonal)
Saturation Signal	50,000 e
Spectral	Monochrome
Peak QE	88% (mono)
Sensor System Noise	7e (at 1 MHz)
Bit Depth	8, 10, or 12 bits
Dynamic Range	78 dB
Sensor Output Sensitivity	12 µV/e
Fill Factor	100%
Frame Rate	CameraLink: 2.9 FPS; FireWire: 2.5 FPS
Gain	0 -36 dB
Binning Frame Rates	2 x 2 at 4.96 fps (strobed) 4 x 4 at 7.68 fps (strobed)
Pixel Clock Rate	12 MHz
Minimum Shutter Speed	2 ms
Lens Adapters	C-mount or F-Mount
Camera Head Dimensions (L x D x H)	74.2 x 91.4 x 95.1 mm with F-Mount adapter 46 x 91.4 x 95.1 mm with C-Mount adapter
Camera Head Weight	625 g
Operating Temperature	0 - 40 ºC, non-condensing
Vibration	10 - 500 Hz, 3G, sinusoidal
Shock	5G

Table 18.8 ES 3200 Camera Head Specifications

18.4.5 ES 2093 Camera Head Specification

Category	Specification
Sensor Model	KAI-2093 with microlens, solid-state
Sensor Type	Interline Transfer Progressive Scan CCD
Pixels	1920 (H) x 1080 (V) 2.074 Megapixels
Pixel Size	7.4 µm x 7.4 µm
Photosensitive Area	14.2 mm (H) x 8 mm (V)
Imager Size	16.3 mm (diagonal)
Spectral	Bayer RGB color or monochrome
Saturation Signal	40,000 e
Bit Depth	8, 10, or 12 bits per color
Dynamic Range	58 dB at full resolution
Peak QE RGB	36%, 33%, 26%
Sensor Output Sensitivity	14 µV/e
Frame Rate	CameraLink: 30 FPS; FireWire: 15.5 FPS monochrome
Maximum Frame Rate	30
Gain	0-36 dB
Binning Frame Rates	2 x 2 @ 52.95 FPS; 3 x 3 @ 69.73 FPS; 4 x 4 @ 90.17 FPS
Pixel Clock Rate	Up to 38 MHz per channel
Min. Integration Time Continuous	94 µS
Max. Integration Time Triggered	60 µS
Max integration time	>5 sec.
Lens Mount	F-Mount, C-Mount
Camera Head Dimensions (L x D x H)	54.62 x 82.8 x 54.62 mm with F-Mount adapter 54.62 x 48.5 x 54.62 mm with C-Mount adapter
Camera Head Weight	258 g
Tether Cable Length	2 m, 5 m, 7 m
Operating Temperature	0 - 40º C, non-condensing
Vibration	10-500 Hz, 3G sinusoidal
Shock	5G

Table 18.9 ES 2093 Camera Head Specifications

18.4.6 ES 2020 Camera Head Specification

Category	Specification
Sensor Model	KAI-2020
Sensor Type	Interline Transfer Progressive Scan CCD
Pixels	1600 (H) x 1200 (V)
Pixel Size	7.4 µm x 7.4 µm
Photosensitive Area	11.8 mm (H) x 8.9 mm (V)
Imager Size	14.78mm (diagonal)
Saturation Signal	40,000 e
Peak QE RGB	41%, 37%, 31%
Sensor Output Sensitivity	30 µV/e
Maximum Frame Rate	30
Lens Adapters	C-mount or F-Mount
Min. integration time in continuous mode	90 µS
Min. integration time in trigger mode	60 µS
Max integration time	>5 sec.
Camera Head Dimensions (L x D x H)	54.62 x 82.8 x 54.62 mm with F-Mount adapter 54.62 x 48.5 x 54.62 mm with C-Mount adapter
Camera Head Weight	309 g with F-Mount adapter 245 g with C-Mount adapter

Table 18.10 ES 2020 Camera Specifications

18.4.7 ES 2001* Camera Head Specification

Category	Specification
Sensor Model	KAI-2001 with microlens, solid-state
Sensor Type	Interline Transfer Progressive Scan CCD
Pixels	1600 (H) x 1200 (V) (1.92 Megapixels)
Pixel Size	7.4 µm x 7.4 µm
Photosensitive Area	11.8 mm (H) x 8.9 mm (V)
Imager Size	14.8 mm (diagonal)
Spectral	RGB Bayer Color or Monochrome
Bit Depth	8, 10, or 12 bits per color
Dynamic Range	60 dB at full resolution
Frame Rate	CameraLink: 30 FPS; FireWire: 15.8 FPS monochrome
Gain	0-36 dB
Binning Frame Rates	2 x 2 @ 52.95 FPS; 3 x 3 @ 69.73 FPS; 4 x 4 @ 82.85 FPS
Pixel Clock Rate	Up to 38 MHz per channel
Min. integration time in continuous mode	90µS
Min. integration time in trigger mode	60µS
Max integration time	>5 sec.
Saturation Signal	40,000 e
Peak QE RGB	45%, 42%, 35%
Sensor System Noise	40 e (at 40 MHz), 23 e (20 MHz)
Sensor Output Sensitivity	16 µV/e
Blooming Suppression	300X
Maximum Frame Rate	31.85
Lens Adapters	C-mount or F-Mount
Tether Cable Length	2 m, 5 m, 7 m
Vibration	10-500 Hz, 3G sinusoidal
Operating Temperature	0 -40 º C, non-condensing
Shock	5G
Camera Head Dimensions (L x D x H)	54.62 x 82.8 x 54.62 mm with F-Mount adapter 54.62 x 48.5 x 54.62 mm with C-Mount adapter
Camera Head Weight	258 g

Table 18.11 ES 2001 Camera Head Specifications

* ES 1602 and ES 2001 have been discontinued.

18.4.8 ES 1602*/1603 Camera Head Specification

Category	Specification
Sensor Model	KAF-1602 / KAF-1603
Sensor Type	Solid-state, Full Frame Progressive Scan CCD
Pixels	1536 H x 1024 V (1.57 Megapixels)
Pixel Size	9.0 µm x 9.0 µm
Photosensitive Area	13.8 mm (H) x 9.2 mm (V)
Imager Size	16.6 mm (diagonal)
Saturation Signal	100,000 e
Spectral	Monochrome
Peak QE	62% (mono)
Bit Depth	8, 10 or 12 bits
Dynamic Range	74 dB
Pixel Clock Rate	12 MHz
Minimum Shutter Speed	2 ms
Sensor System Noise	15 e (at 10 MHz)
Sensor Output Sensitivity	10 µV/e
Fill Factor	100%
Maximum Frame Rate	6.42*
Lens Adapters	F-Mount or C-mount
Binning	Yes
Camera Head Dimensions (L x D x H)	91.4 x 74.2 x 91.5 mm with F-Mount adapter 91.4 x 46 x 95.1 mm with C-Mount adapter
Camera Head Weight	625 g
Operating Temperature	0 - 40 ºC, non-condensing
Vibration	10 - 500 Hz, 3G, sinusoidal
Shock	5G

*Does not include integration or shutter time.

Table 18.12 ES 1602/1603 Camera Head Specifications

* ES 1602 and ES 2001 have been discontinued

18.4.9 Multi-Head Controller Specification

Category	Specifications
Camera Head Inputs	4
Controller Data Interface	CameraLink Serial, IEEE 1394a RS232
Image Data Interface	CameraLink (medium) IEEE 1394a
CameraLink Ports	Base and Medium/Dual Base
Output bit Depth	8, 10, or 12 per channel
Trigger Ports	BNC (trigger in, strobe output)
Output Video Ports	DVI
Controller Dimensions (L x D x H)	157 x 157 x 50.8 mm
Controller Weight	1140 g
Operating Temperature	0 - 40 °C, non-condensing
Vibration	3G, sinusoidal from 10 to 500 Hz
Shock	5G
CE Certification	Yes
Input Voltage	12 VDC
Trigger Input Voltage (BNC)	4 -20 VDC
Strobe Output	Switches between 0 VDC and 5 VDC with 3K of series resistance
Power Consumption	12 W to 16 W (varies with head model and number of attached heads)
Power Connector	LEMO elbow receptacle, EPG.OB.302. HLN for 12 VDC power input. This receptacle mates with the Fgg.OB.302.CLAD56 plug. Pin 1 is 12 VDC and Pin 2 is connected to ground.

Table 18.13 Multi-Head Controller Specifications

18.4.10 Single-Head Controller Specification

Category	Specifications
Camera Head Inputs	1
Controller Data Interface	CameraLink Serial, RS232
Image Data Interface	CameraLink (medium)
CameraLink Ports	Base and Medium
Output bit Depth	8, 10, or 12 per channel
Trigger Ports	BNC (trigger in, strobe output)
Controller Dimensions (L x D x H)	158.76 x 152.04 x 38.11 mm
Controller Weight	992 g
Operating Temperature	0 - 40 °C, non-condensing
Vibration	3G, sinusoidal from 10 to 500 Hz
Shock	5G
CE Certification	Yes
Input Voltage	Approximately 12 VDC depending on camera head model
Trigger Input Voltage (BNC)	4 -20 VDC
Strobe Output	Switches between 0 VDC and 5 VDC with 3K of series resistance
Power Consumption	12 W to 16 W (varies with head model and number of attached heads)
Power Connector	LEMO elbow receptacle, EPG.OB.302. HLN for 12 VDC power input. This receptacle mates with the Fgg.OB.302.CLAD56 plug. Pin 1 is 12 VDC and Pin 2 is connected to ground.

Table 18.14 Single-Head Controller Specifications

18.4.11 GigE Vision Single-Head Controller Specification

Category	Specifications
Camera Head Inputs	1
Interface	100/1000 Base (GigE Vision compliant)
Serial	RS232
Diagnostics	10/100/1000 Base-T (Diagnostics and Firmware upgrade only)
Output Bit Depth	8, 10, or 12 per channel
Trigger Ports	BNC, TTL (trigger in, strobe output)
Controller Dimensions (L x D x H)	158.8 x 152.0 x 50.4 mm
Controller Weight	1124 g
Operating Temperature	0 - 40 °C, non-condensing
Vibration	3G, sinusoidal from 10 to 500 Hz
Shock	5G
CE Certification	Yes
Power Input	12 VDC, 3 A (max.)
Strobe Output	Switches between 0 VDC and 5 VDC with 3K of series resistance
Power Consumption (Camera and Controller)	15 Watts
Power Connector	LEMO elbow receptacle, EPG.OB.302. HLN for 12 VDC power input. This receptacle mates with the Fgg.OB.302.CLAD56 plug. Pin 1 is 12 VDC and Pin 2 is connected to ground.

Table 18.15 GigE Vision Single-Head Controller Specifications

18.5 Sensor Quantum Efficiencies

18.5.1 KAI-16000 Sensor Quantum Efficiencies

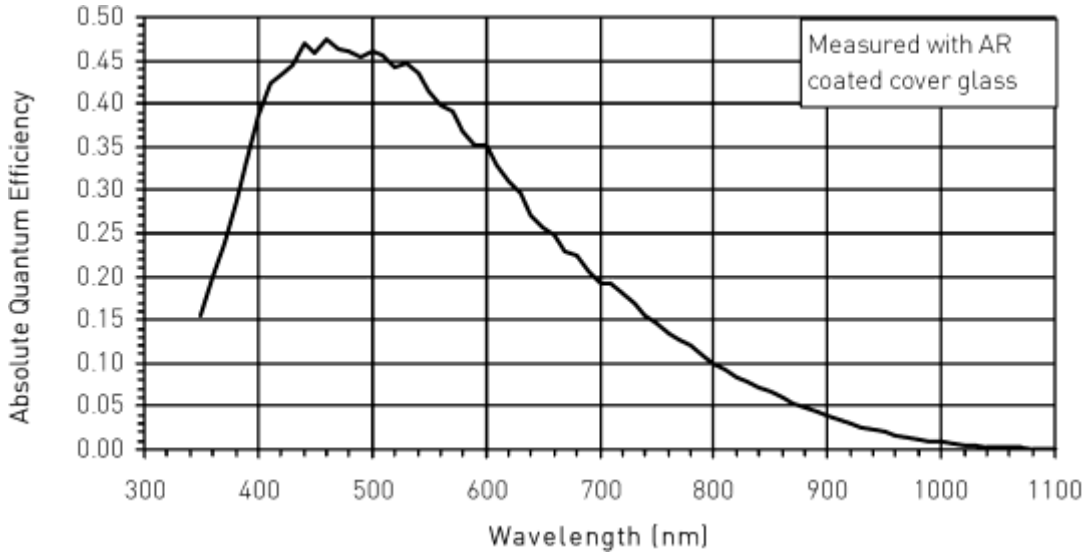


Figure 18.1 KAI-16000 Mono Quantum Efficiency

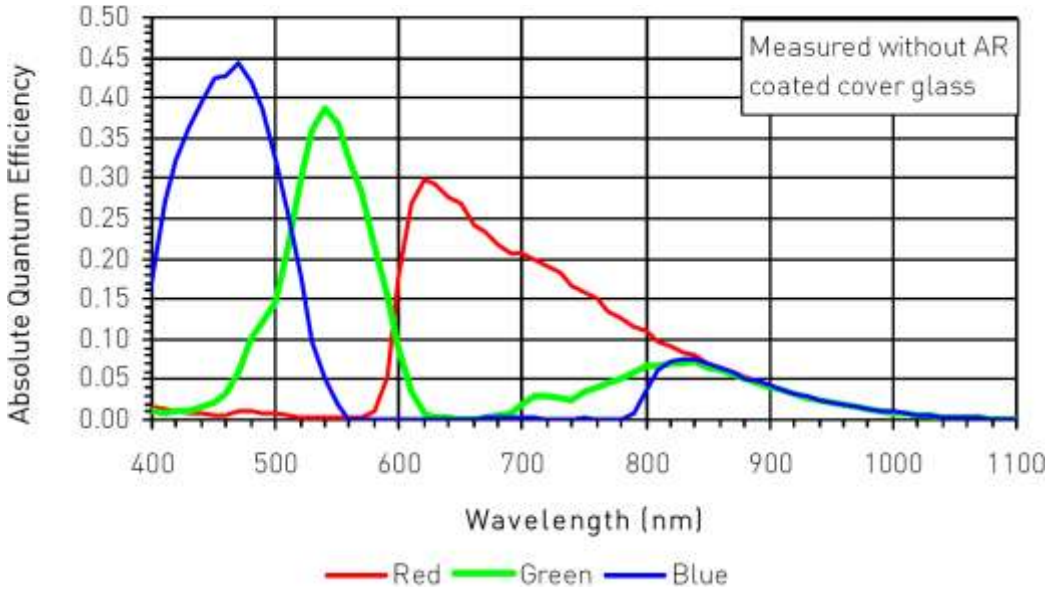


Figure 18.2 KAI-16000 Color Quantum Efficiency

18.5.2 KAI-11002 Sensor Quantum Efficiencies

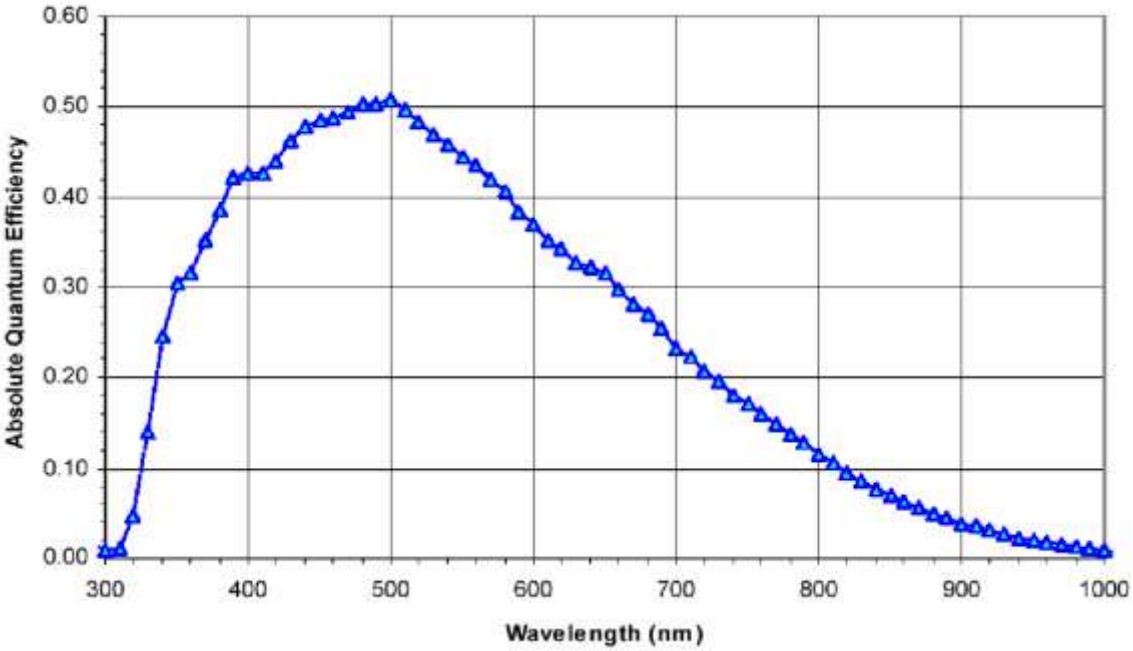


Figure 18.3 KAI-11002 Mono Quantum Efficiency

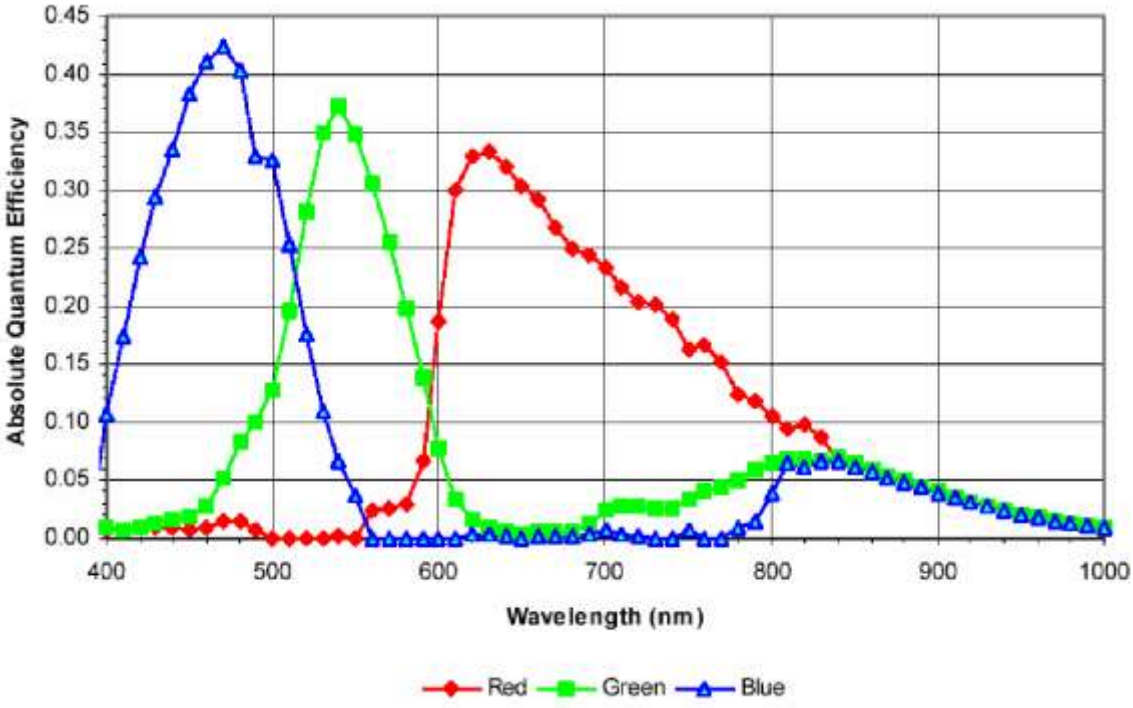


Figure 18.4 KAI-11002 Color Quantum Efficiency

18.5.3 KAI-4021 Sensor Quantum Efficiencies

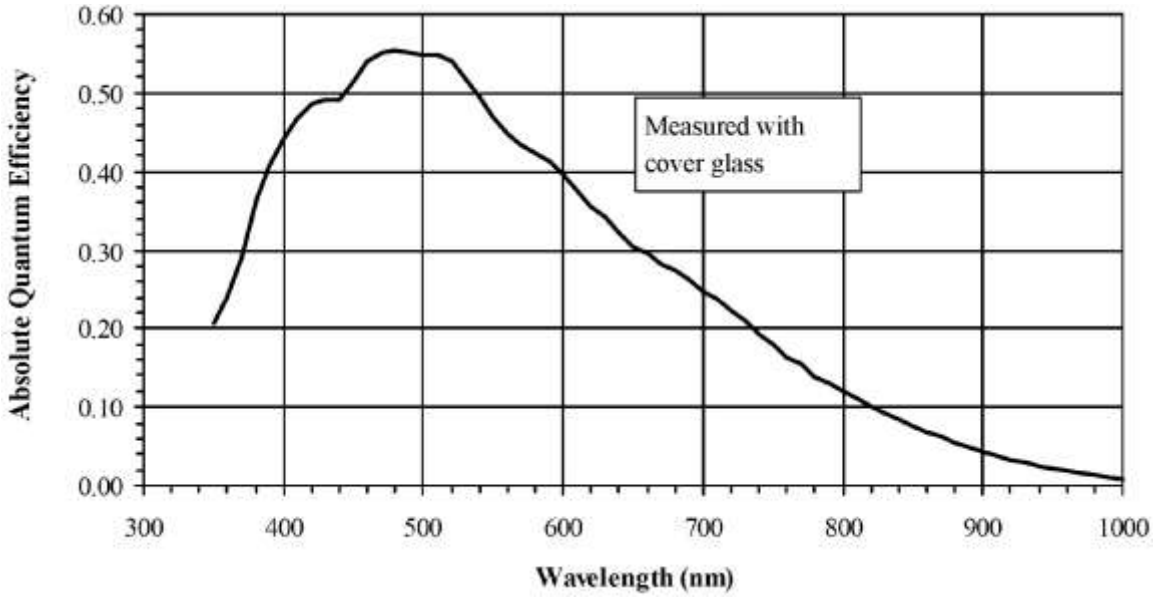


Figure 18.5 KAI-4021 Mono Quantum Efficiency

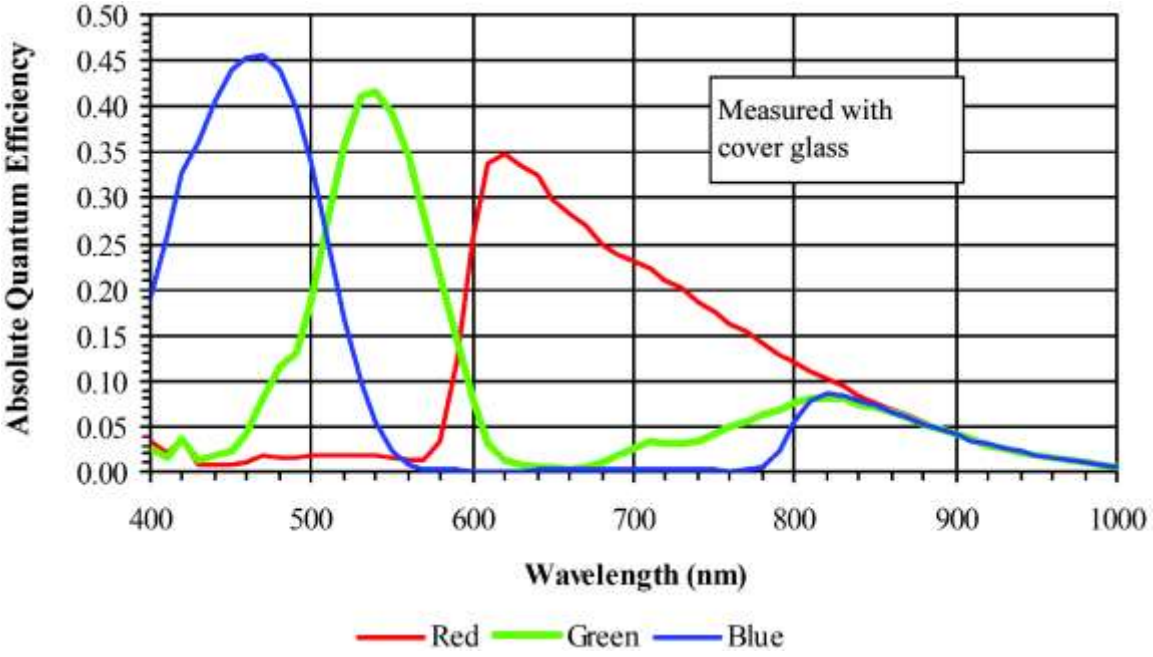


Figure 18.6 KAI-4021 Color Quantum Efficiency

18.5.4 KAF 3200 ME Sensor Quantum Efficiencies

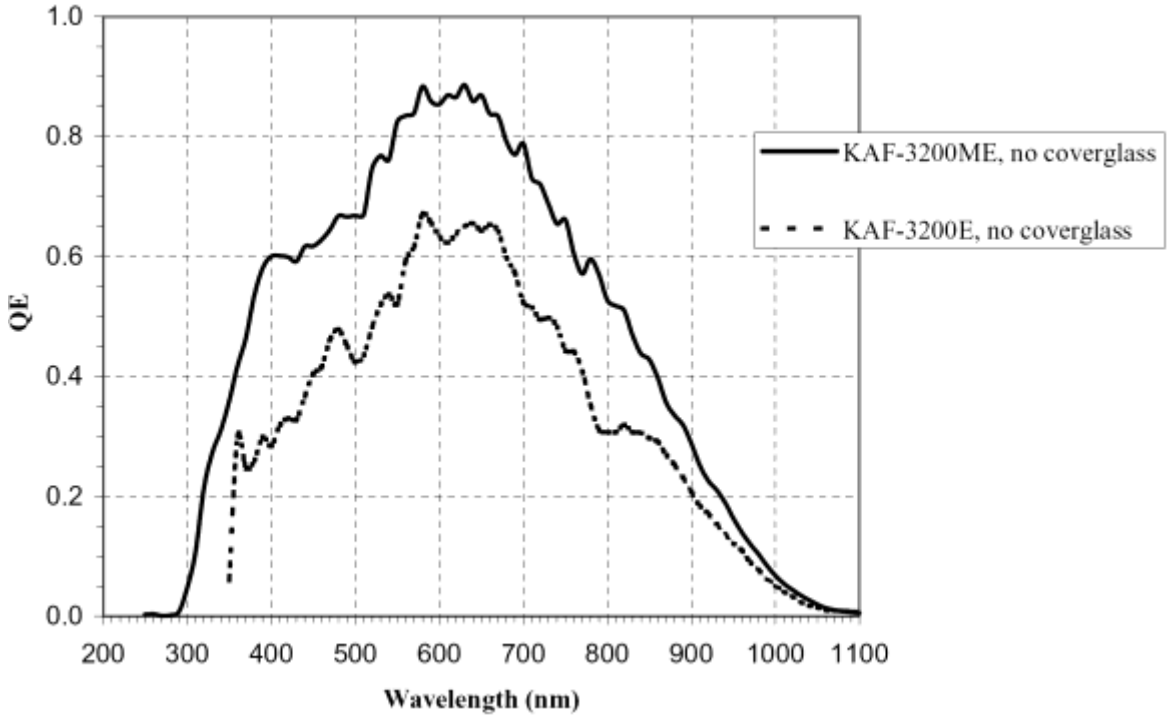


Figure 18.7 KAF 3200 ME Mono Quantum Efficiency

18.5.5 KAI-2093 Sensor Quantum Efficiencies

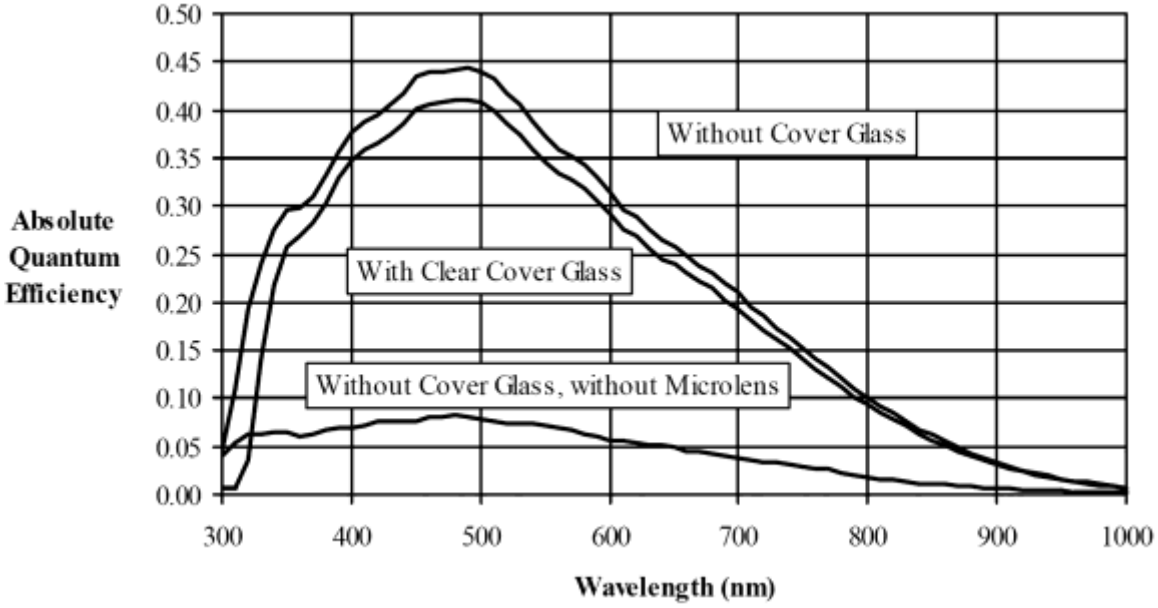


Figure 18.8 KAI-2093 Mono Quantum Efficiency

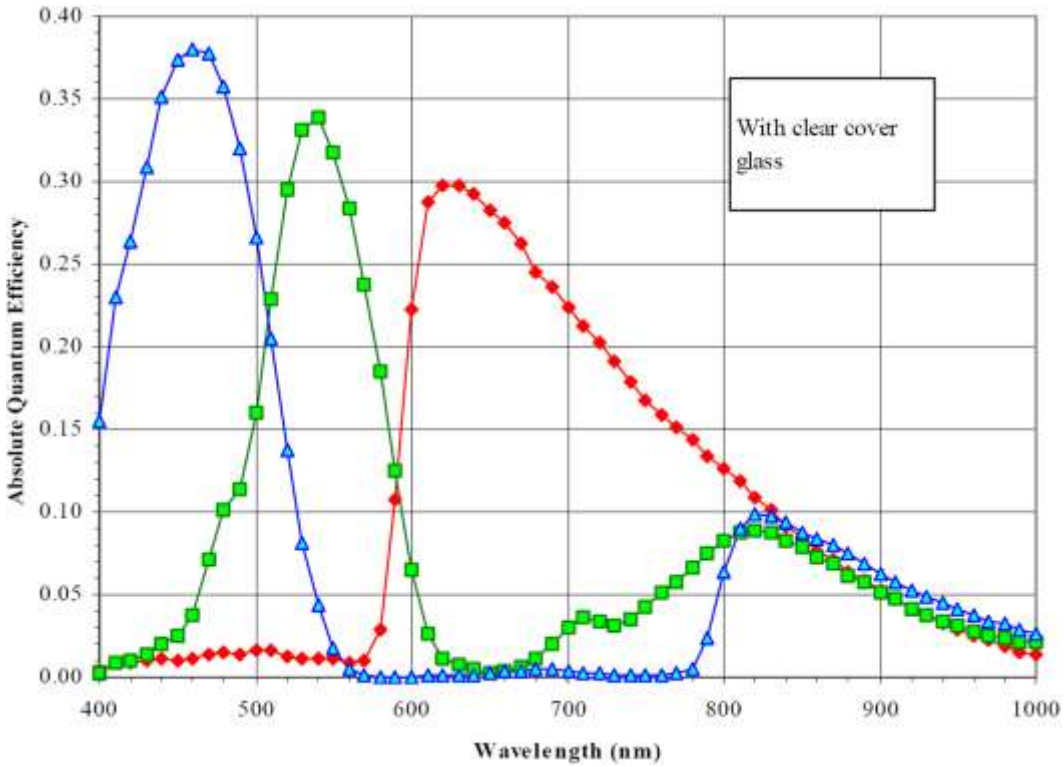


Figure 18.9 KAI-2093 Color Quantum Efficiency

18.5.6 KAI-2020 Sensor Quantum Efficiencies

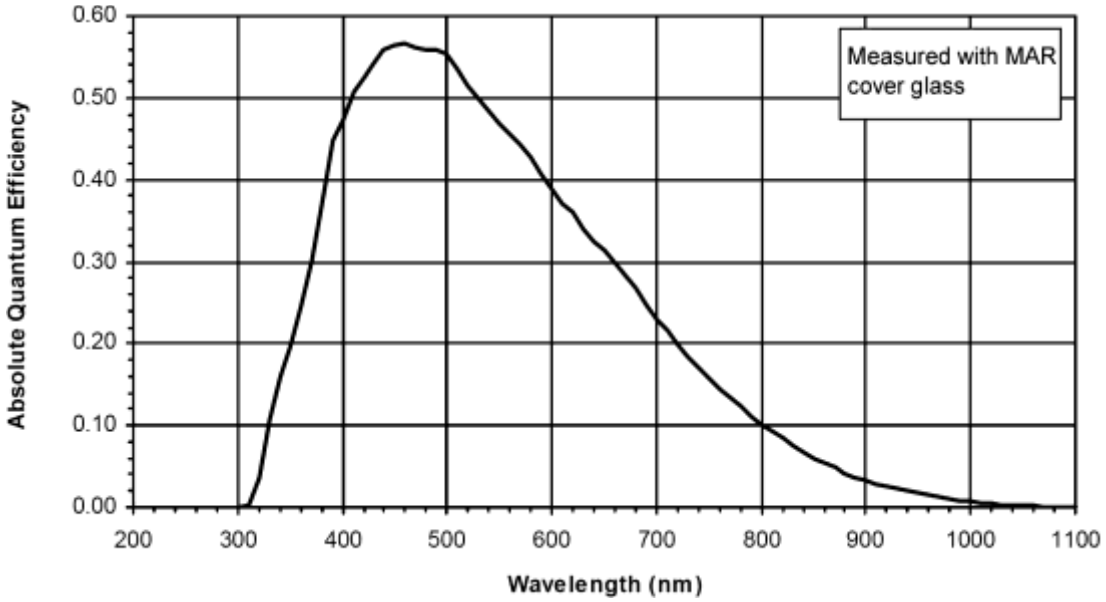


Figure 18.10 KAI-2020 Mono Quantum Efficiency

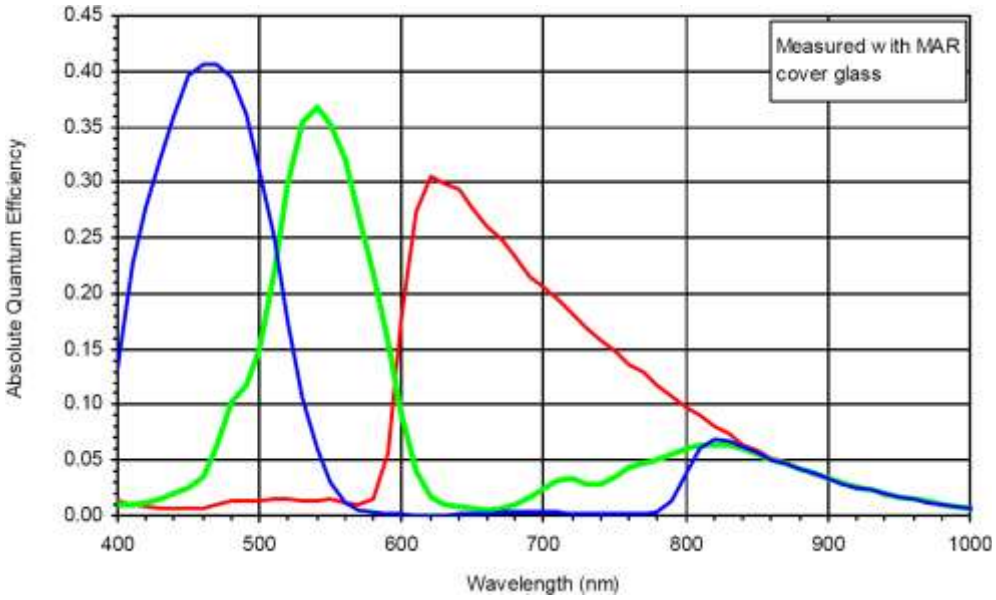


Figure 18.11 KAI-2020 Color Quantum Efficiency

18.5.7 KAI-2001 Sensor Quantum Efficiencies

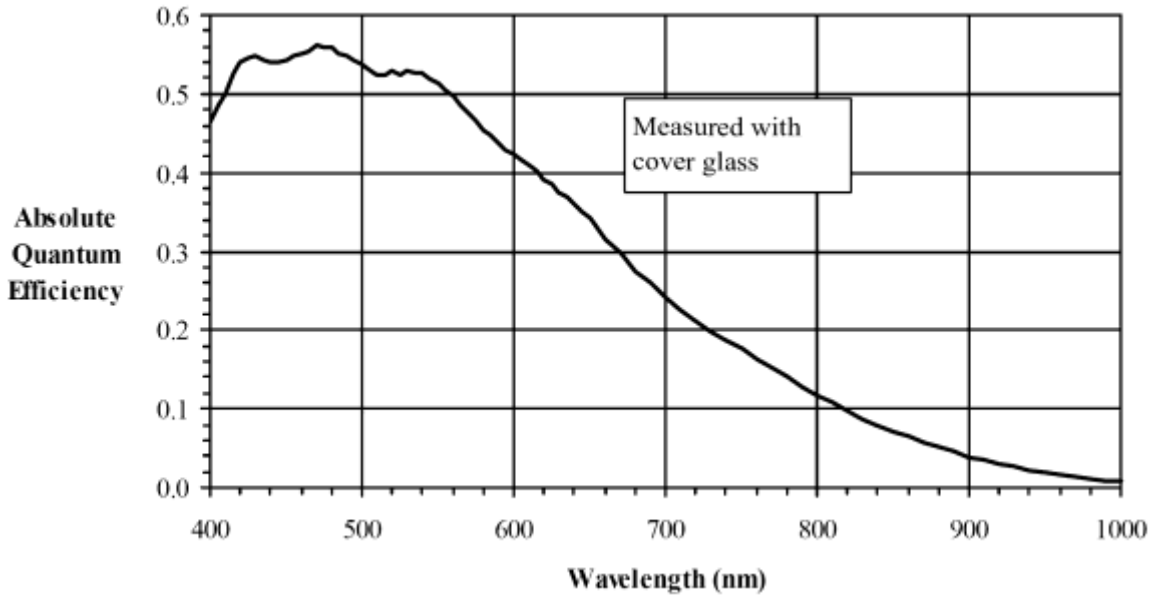


Figure 18.12 KAI-2001 Mono Quantum Efficiency

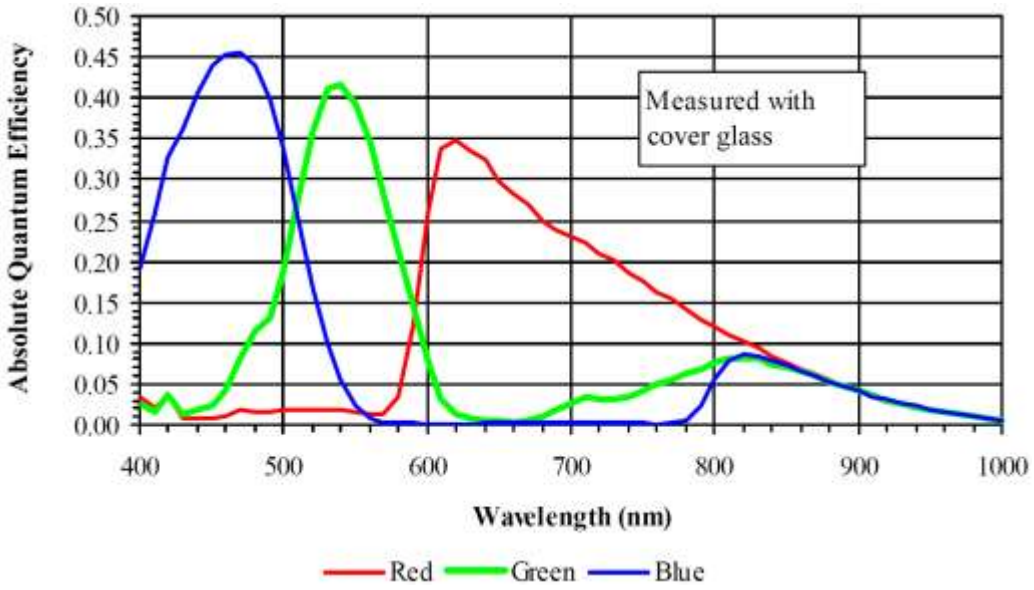


Figure 18.13 KAI-2001 Color Quantum Efficiency

18.5.8 KAF 1602/1603 Sensor Quantum Efficiencies

Figure 18.14 KAF 1602 Quantum Efficiency

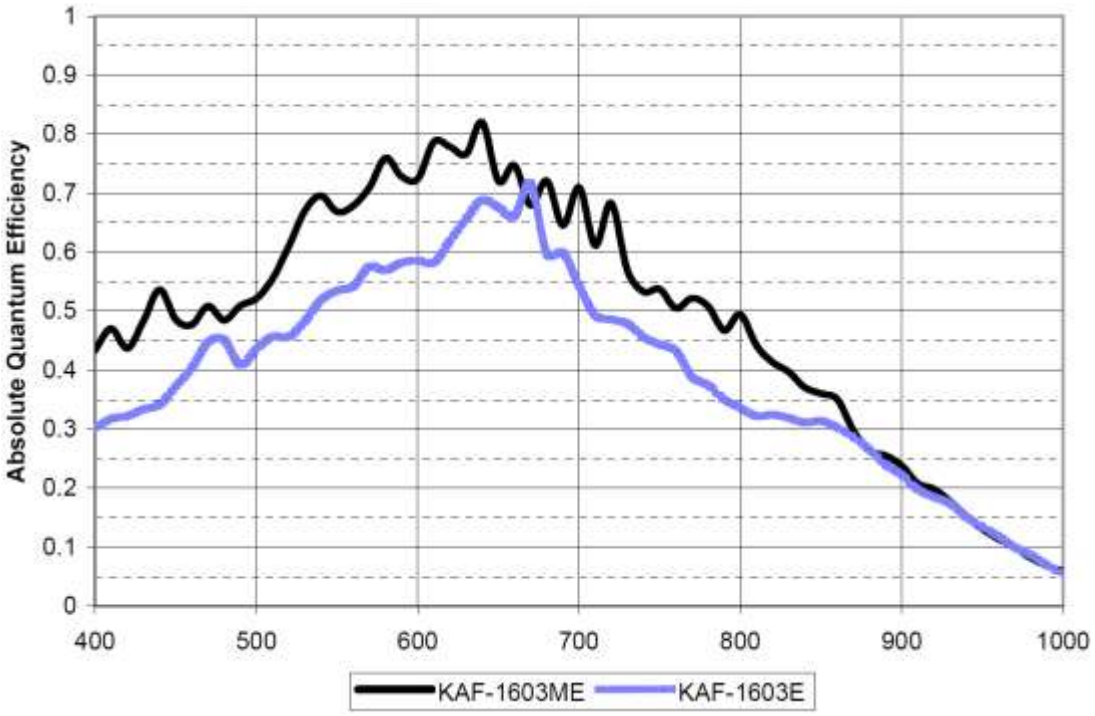


Figure 18.15 KAF 1603 Quantum Efficiency

18.6 Camera Component Mechanical Dimensions

18.6.1 EC 16000/11000 Camera Head Mechanical Dimensions

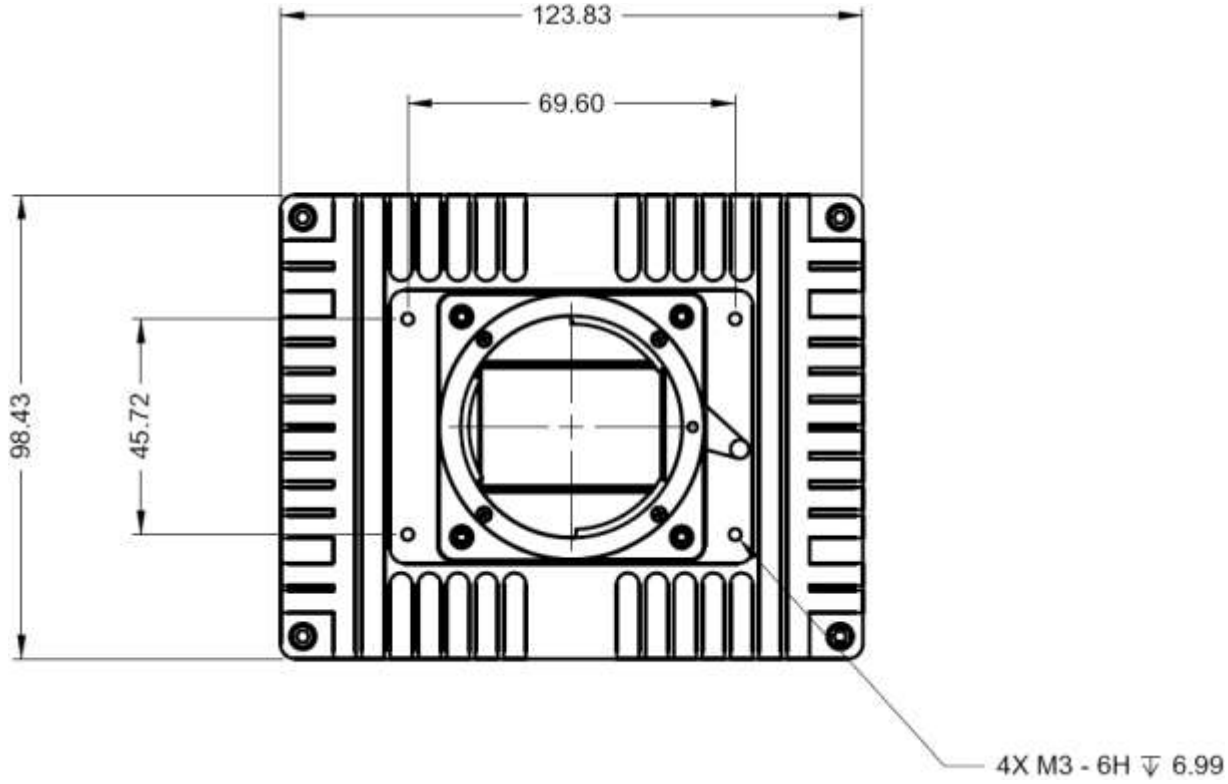


Figure 18.16 EC 16000/11000 Mechanical Dimensions Front View with F-mount

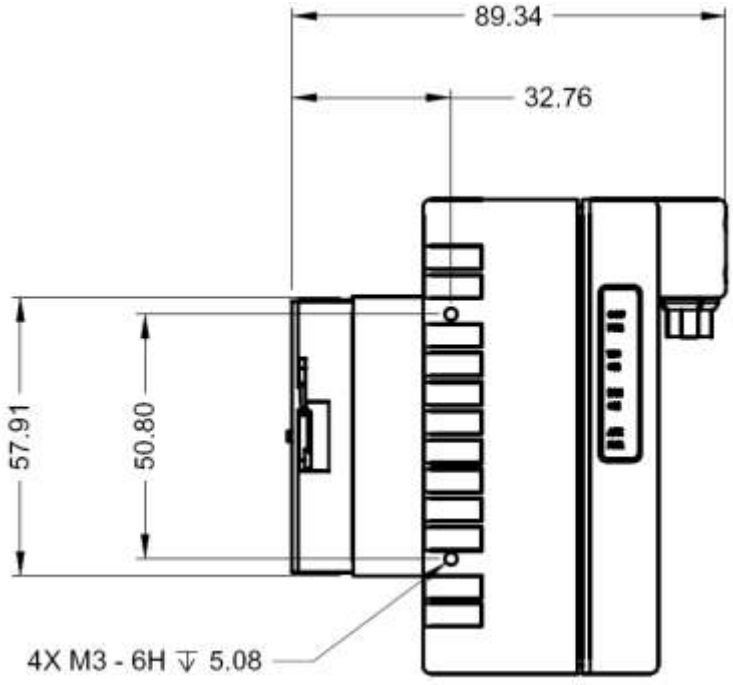


Figure 18.17 EC 16000 & EC 11000 Mechanical Dimensions Side View with F-mount

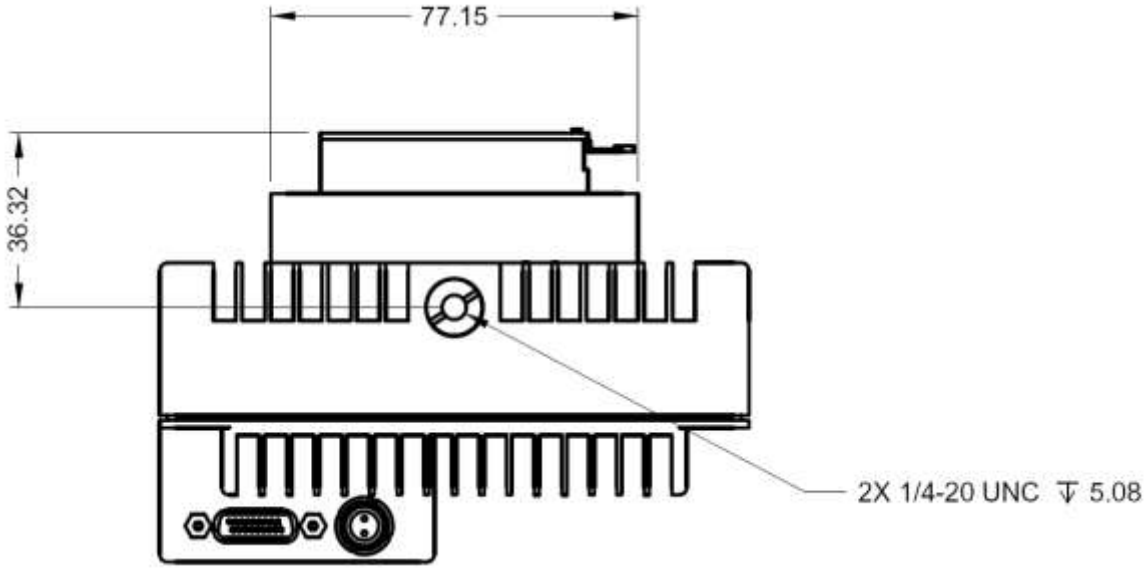


Figure 18.18 EC 16000/11000 Mechanical Dimensions Bottom View with F-mount

18.6.2 EM 11000 Camera Head Mechanical Dimensions

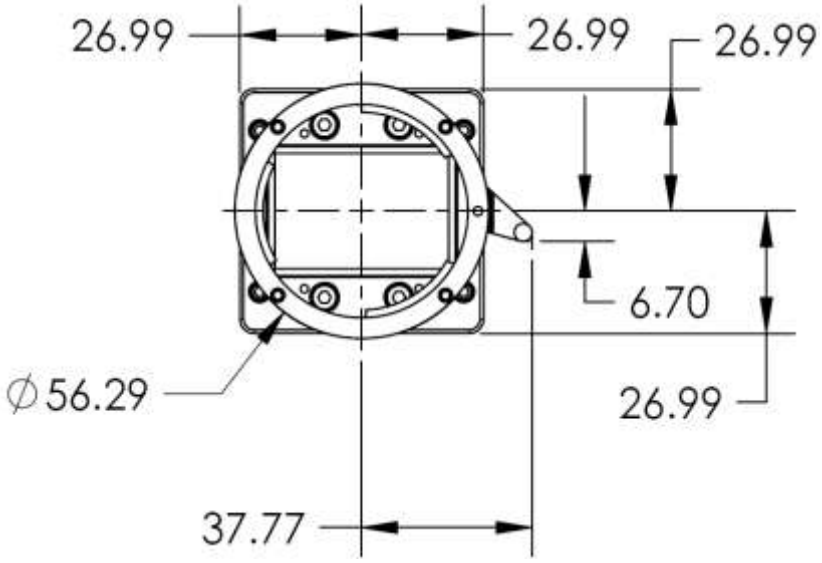


Figure 18.19 EM 11000 Mechanical Dimensions Front View with F-mount

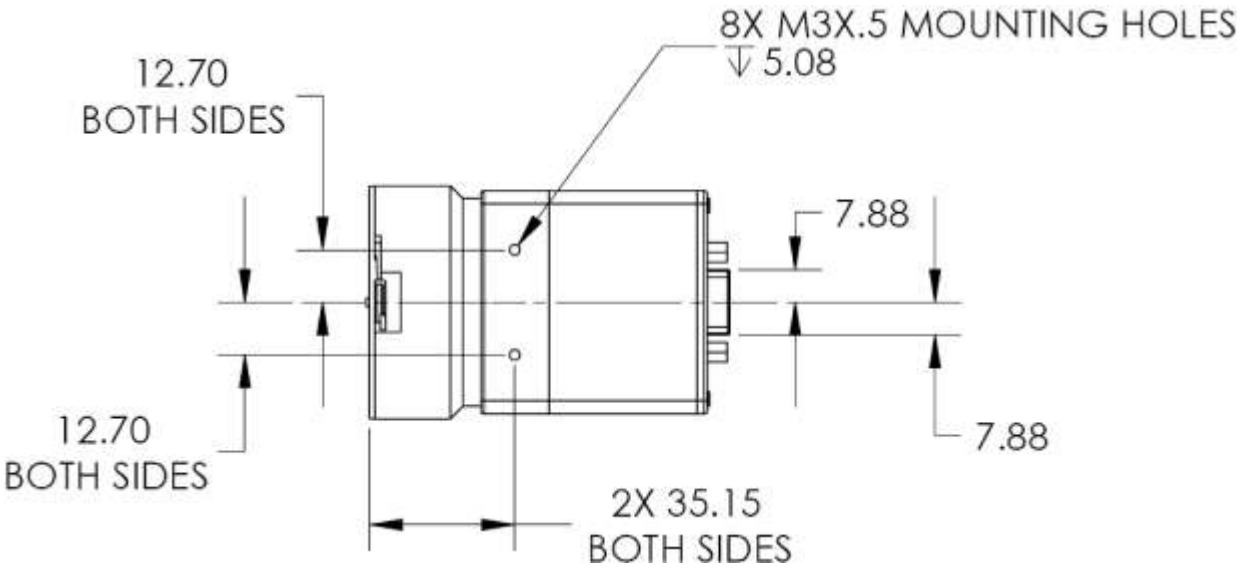


Figure 18.20 EM 11000 Mechanical Dimensions Side View with F-mount

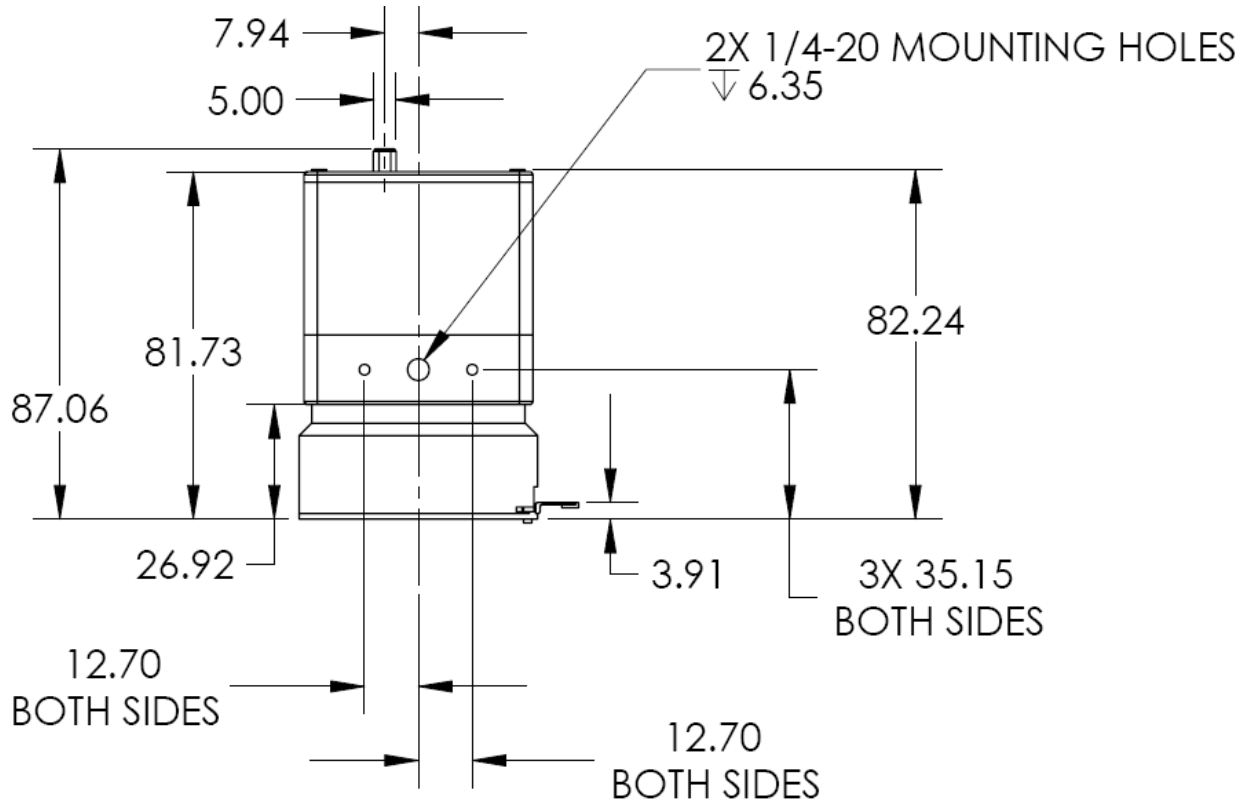


Figure 18.21 EM 11000 Mechanical Dimensions Top View with F-mount

18.6.3 EP 16000/11000 Camera Head Mechanical Dimensions

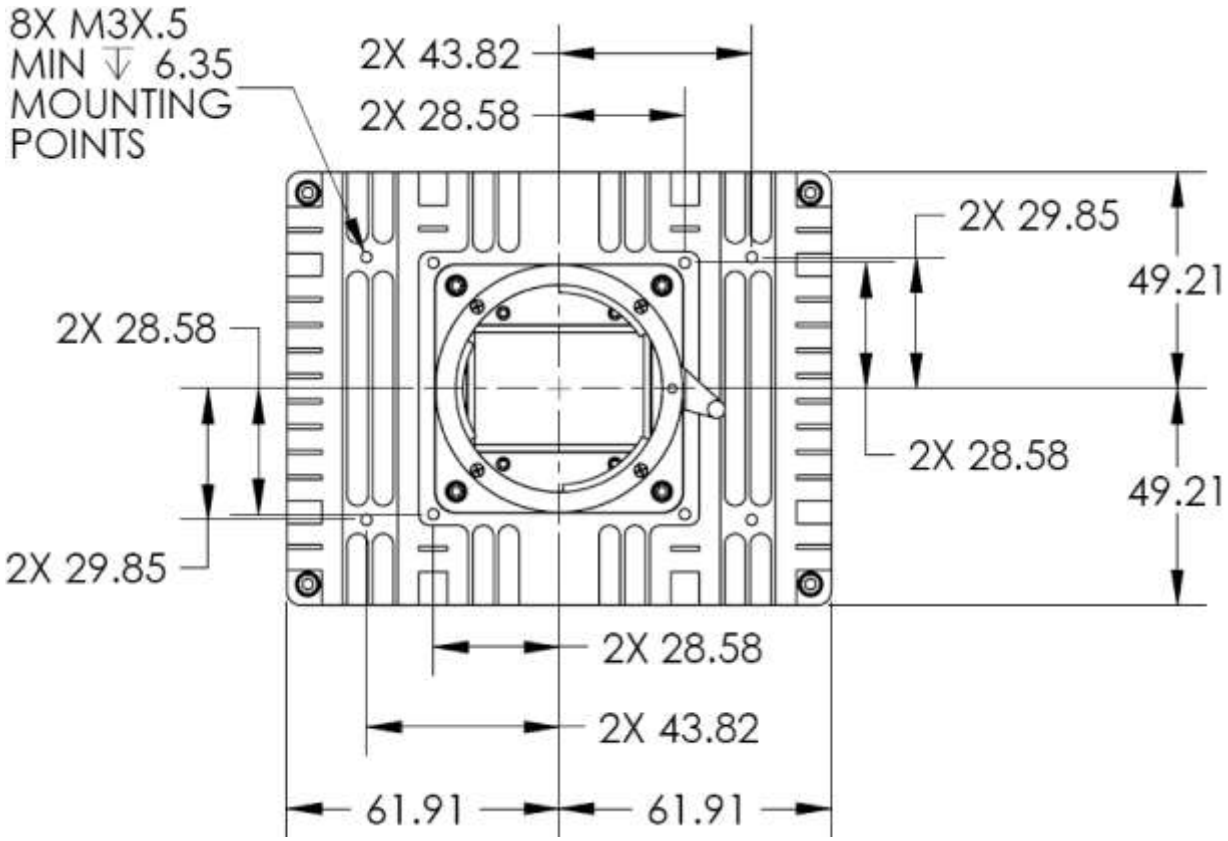


Figure 18.22 EP 16000/11000 Mechanical Dimensions Front View with F-mount

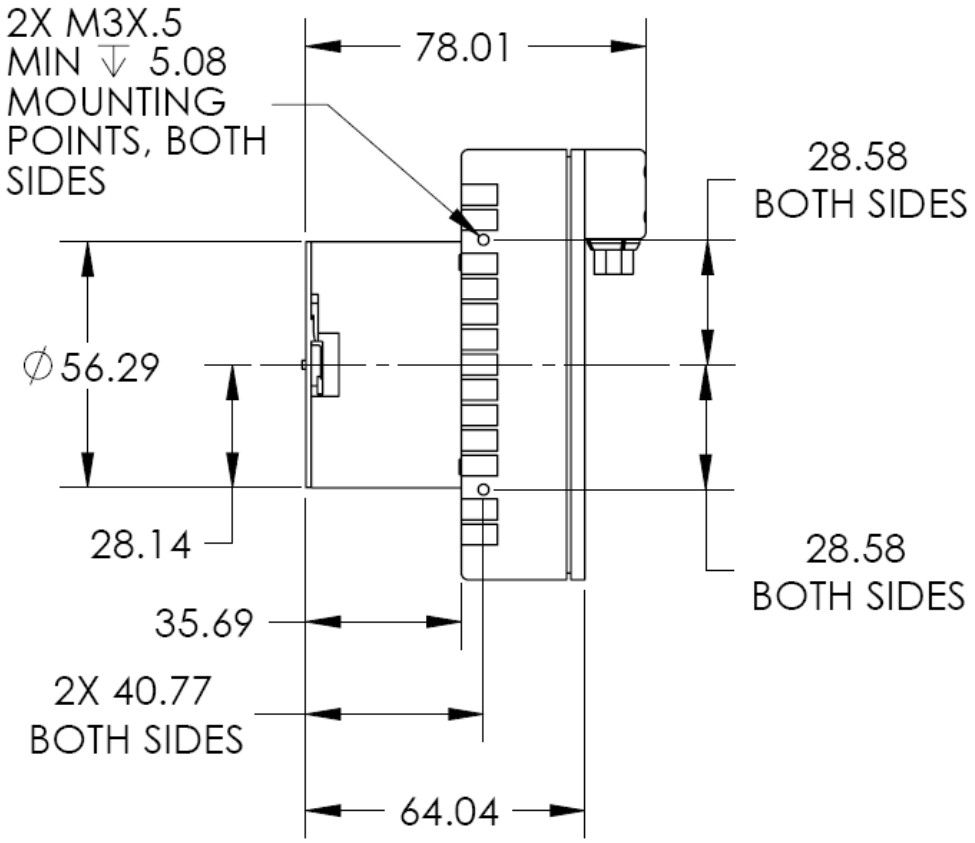


Figure 18.23 EP 16000/11000 Mechanical Dimensions Side View with F-mount

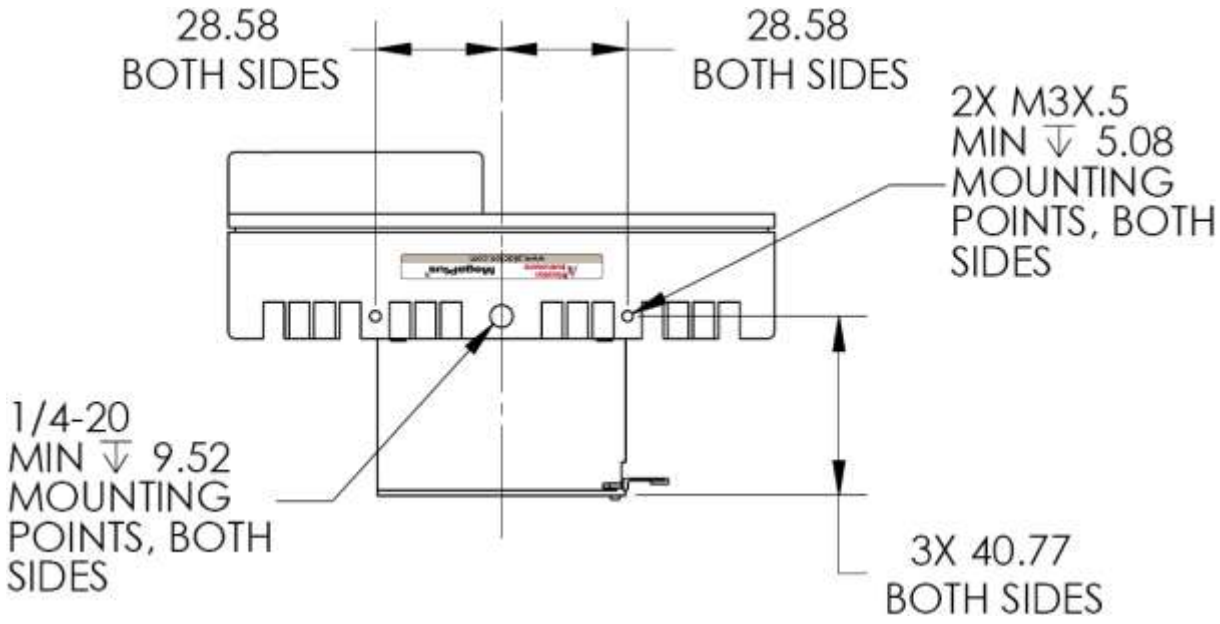


Figure 18.24 EP 16000/11000 Mechanical Dimensions Top View with F-mount

18.6.4 ES 11000 Camera Head Mechanical Dimensions

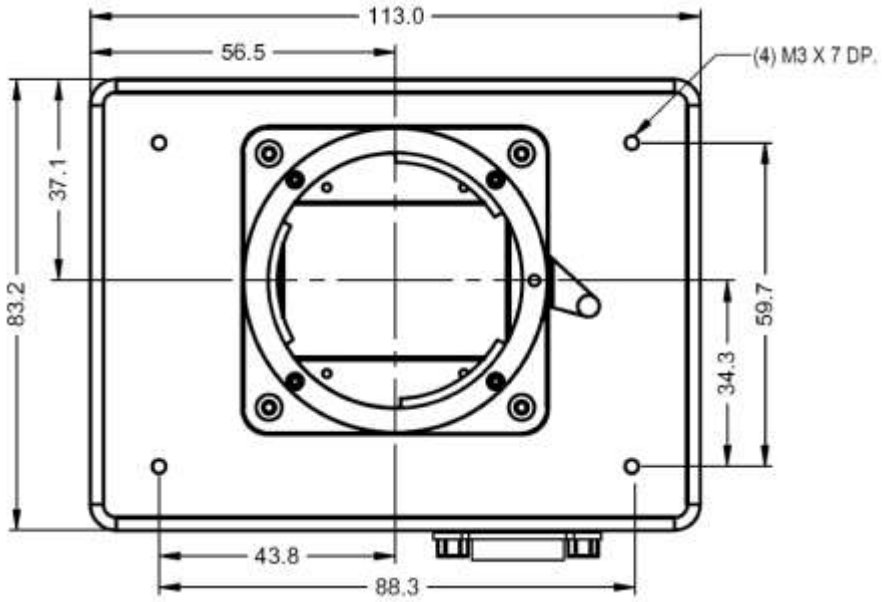


Figure 18.25 ES 11000 Mechanical Dimensions Front View with F-mount and Bottom Plug

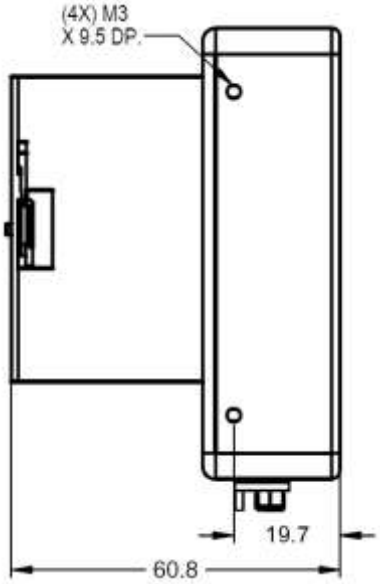


Figure 18.26 ES 11000 Mechanical Dimensions Side View with F-mount and Bottom Plug

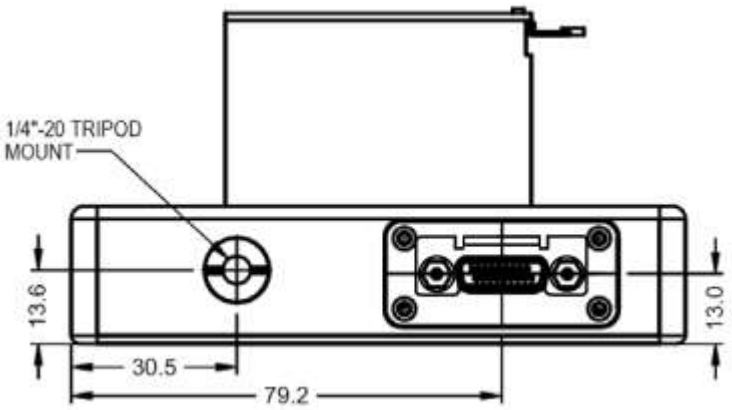


Figure 18.27 ES 11000 Mechanical Dimensions Bottom View with F-mount and Bottom Plug

18.6.5 ES 4020/2093/2020/2001* Camera Head Mechanical Dimensions

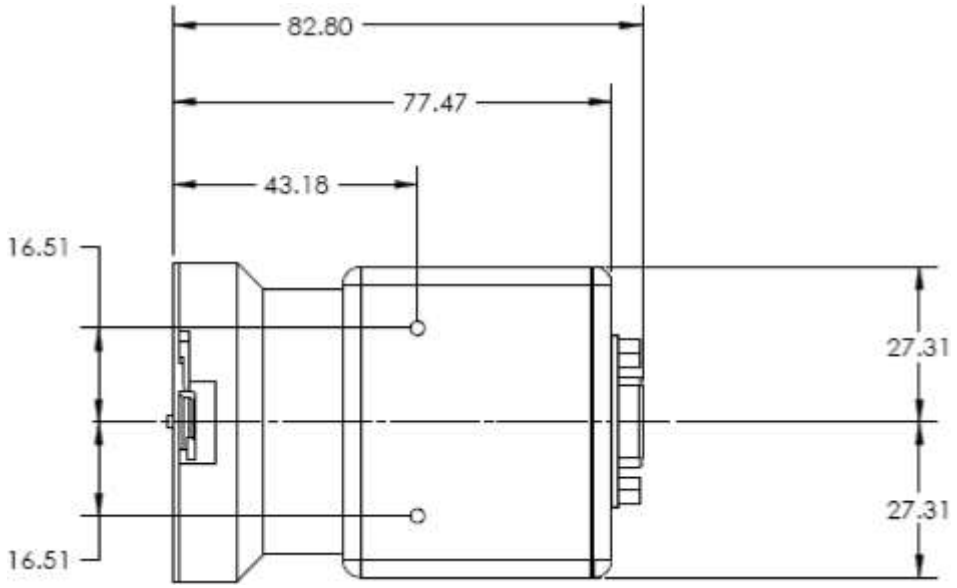


Figure 18.28 ES 4020/2093/2020/2001 Mechanical Dimensions Side View with F-mount

* ES 2001 has been discontinued.

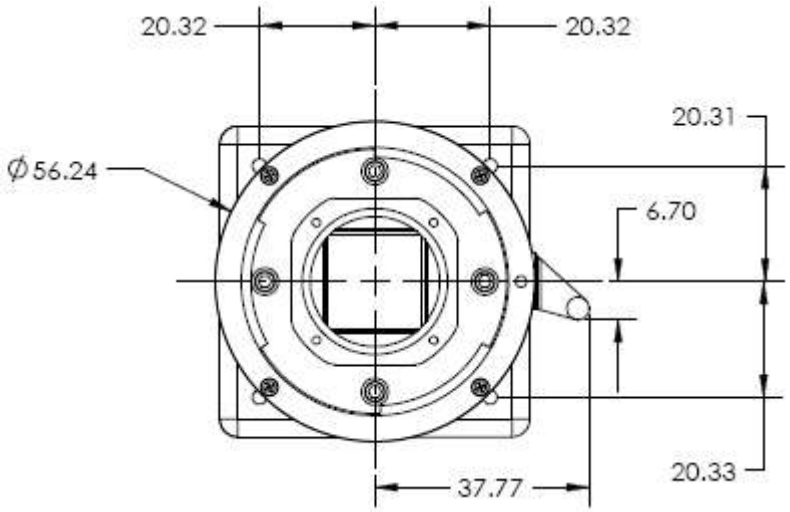


Figure 18.29 ES 4020/2093/2020/2001 Mechanical Dimensions Front View with F-mount

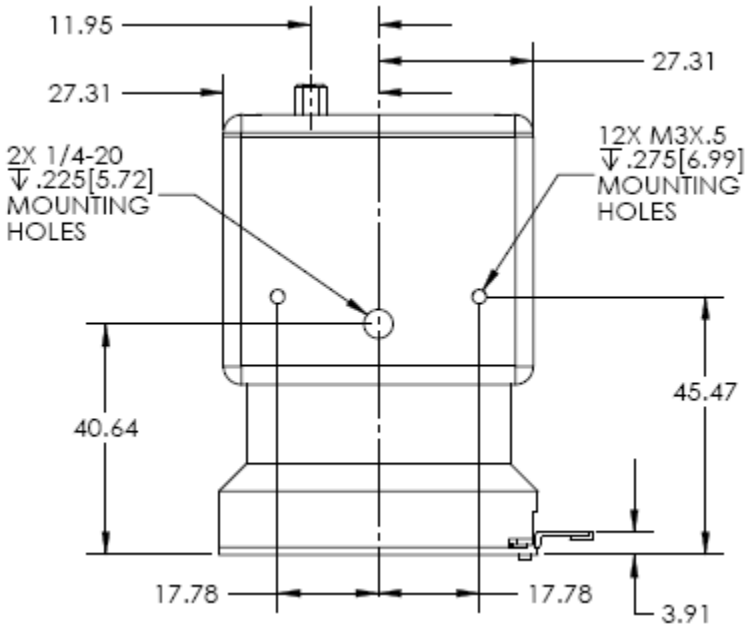


Figure 18.30 ES 4020/2093/2020/2001 Mechanical Dimensions Top View with F-mount

18.6.6 ES 3200/1603/1602* Camera Head Mechanical Dimensions

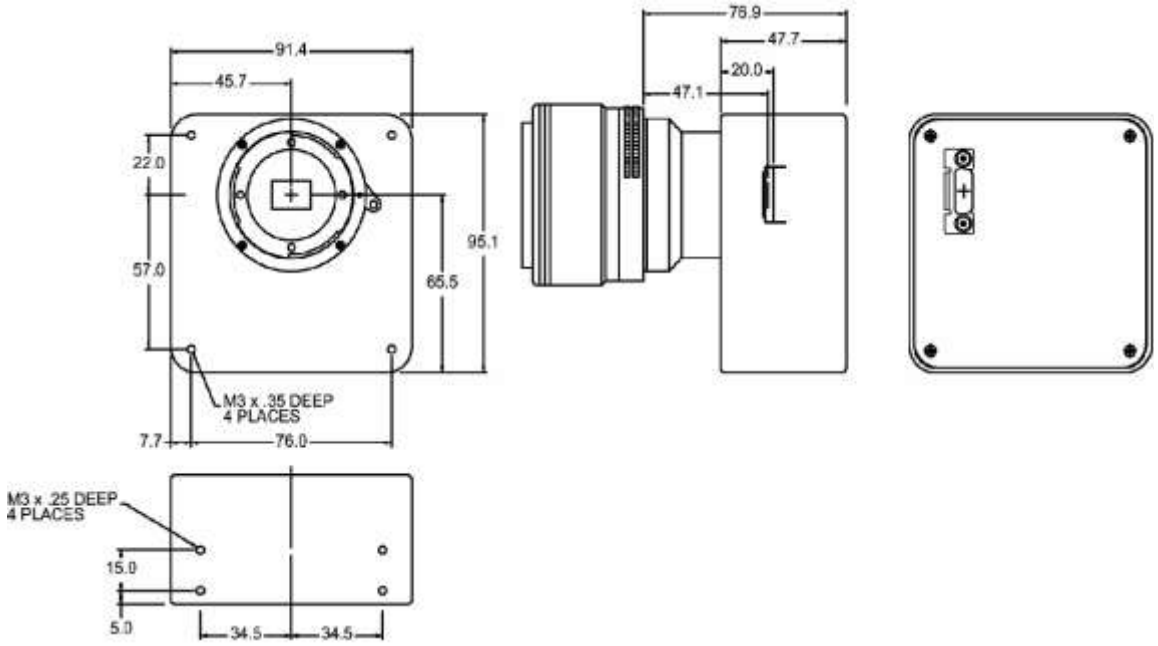


Figure 18.31 ES 3200/1603/1602 Mechanical Dimensions with F-mount

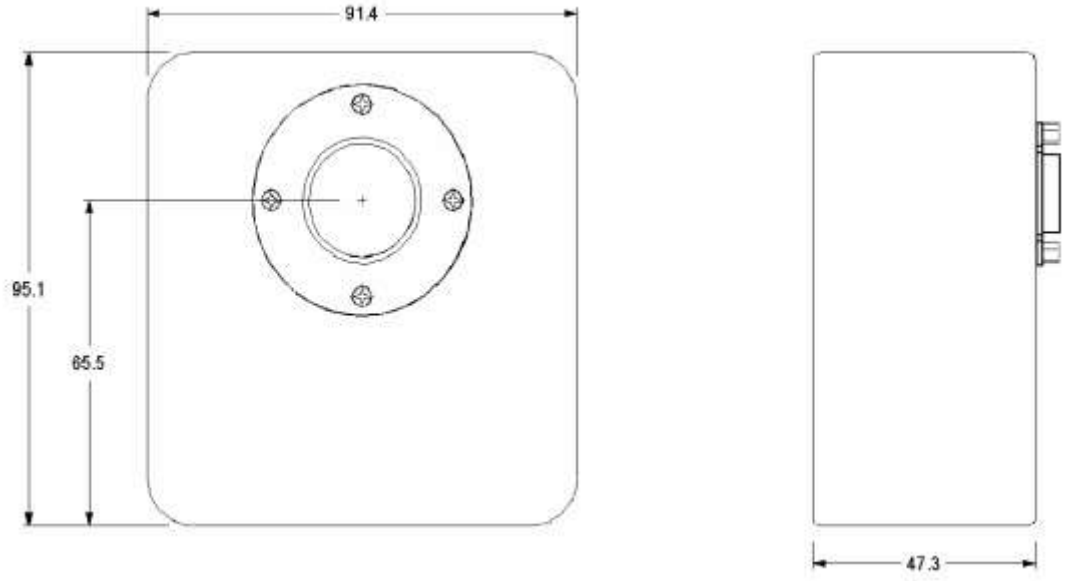


Figure 18.32 ES 3200/1603/1602 Mechanical Dimensions with C-mount

* ES 1602 has been discontinued.

18.6.7 Multi-Head Controller Mechanical Dimensions

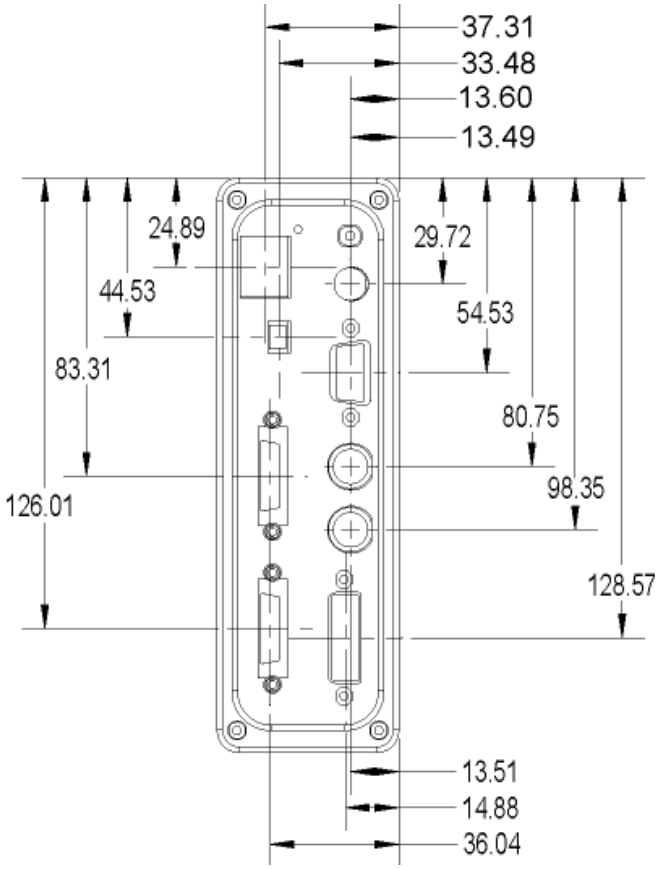


Figure 18.33 Multi-Head Controller Rear View, Rotated 90°

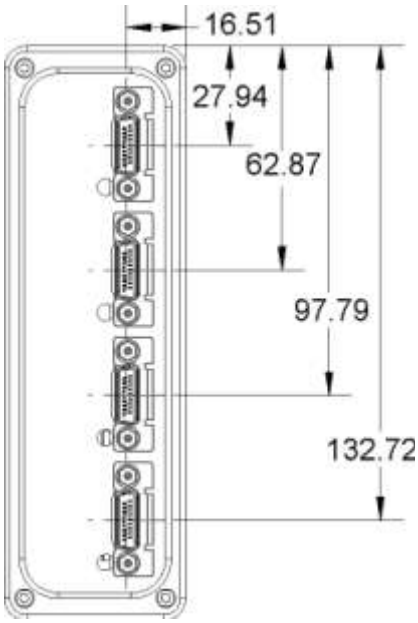


Figure 18.34 Multi-Head Controller Front View, Rotated 90°

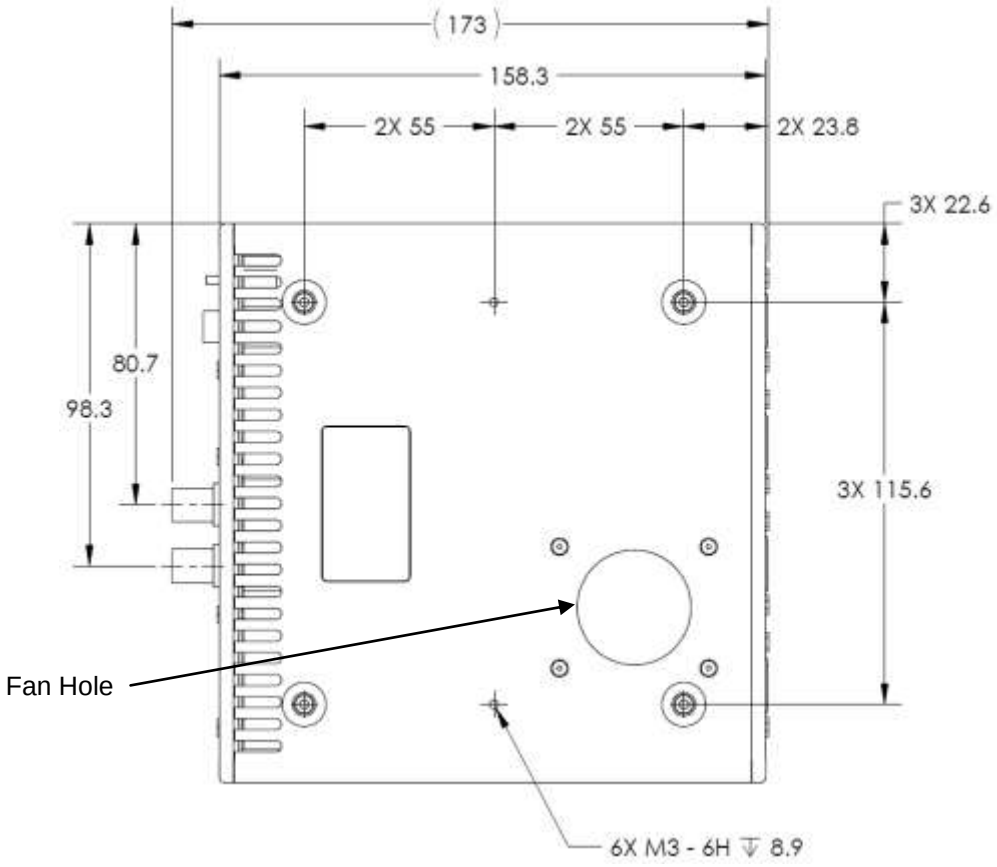


Figure 18.35 Multi-Head Controller Bottom View, Rotated 90°

18.6.8 Single-Head Controller Mechanical Dimensions

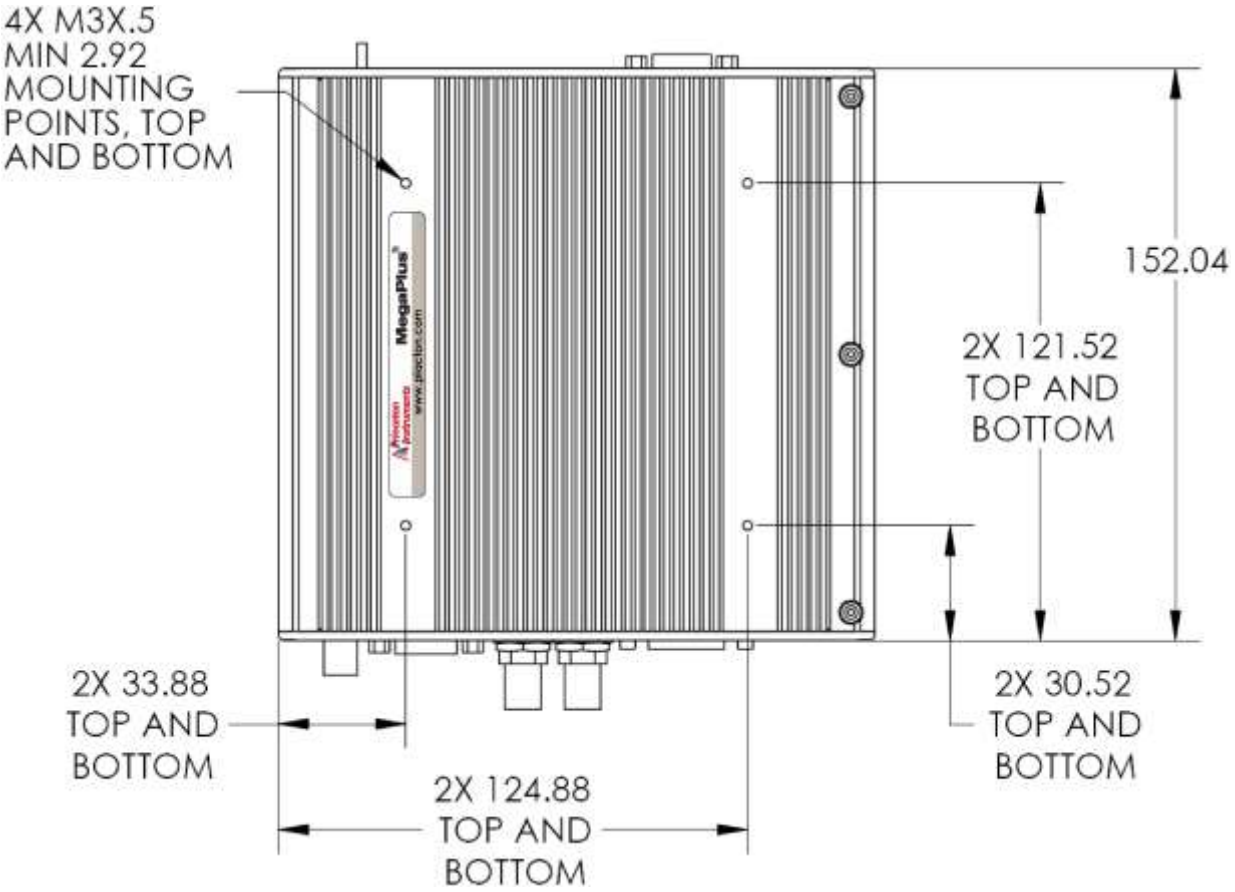


Figure 18.36 Single-Head Controller Top View

NOTE: Single-Head Controller top and bottom dimensions and mounting holes are identical if rubber feet are removed. Controllers are designed to fit together when stacked vertically.

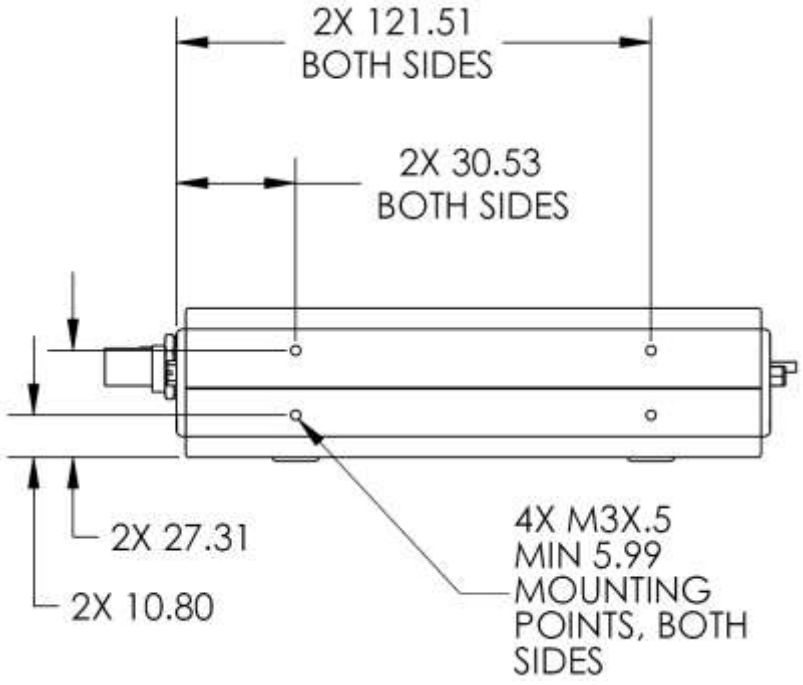


Figure 18.37 Single-Head Controller Side View

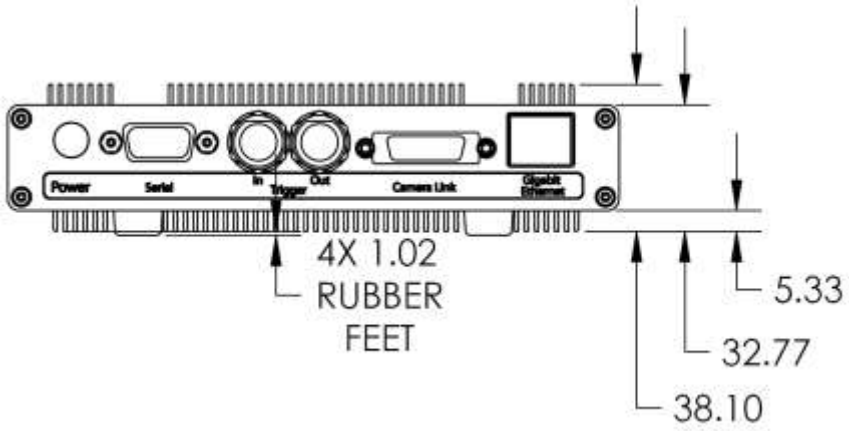
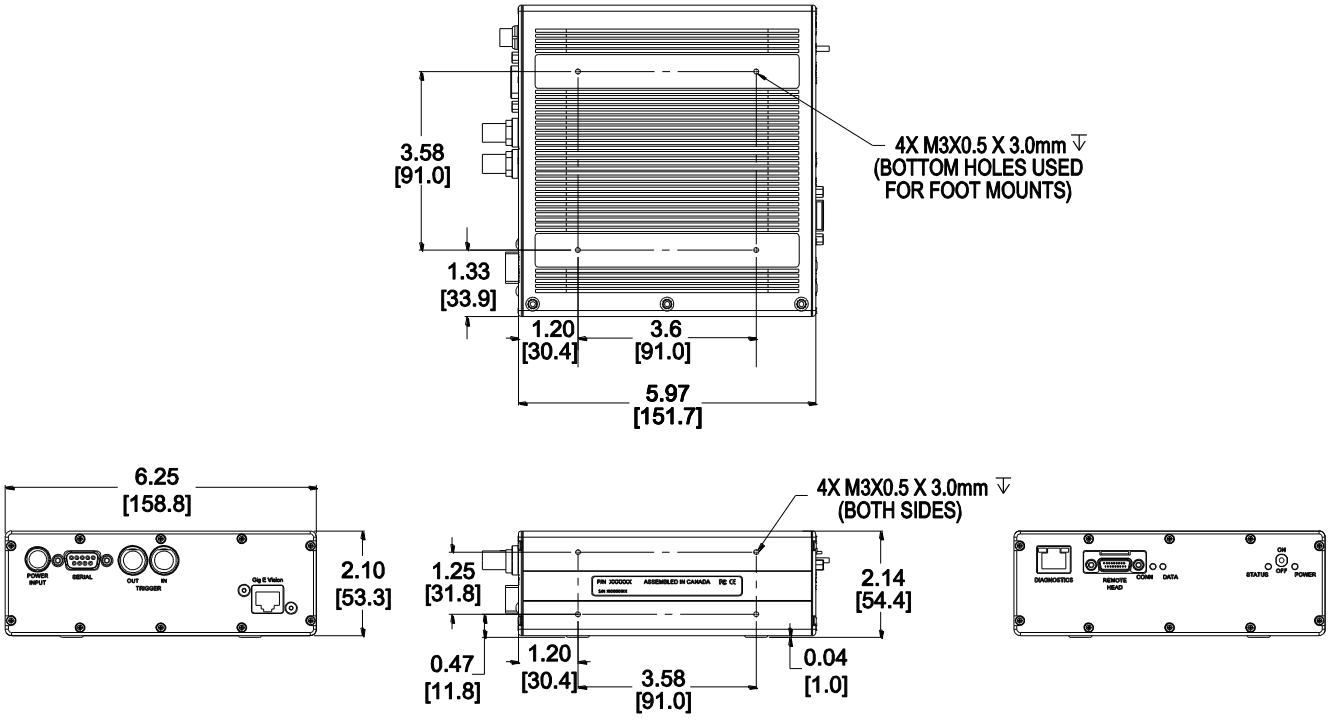


Figure 18.38 Single-Head Controller Rear View

18.6.9 GigE Vision Single-Head Controller Mechanical Dimensions



All dimensions are in inches [mm]

Figure 18.39 GigE Vision Single-Head Controller

19. Maintenance, Technical Support and Warranty

19.1 Maintenance

There are no user serviceable parts inside the camera. The camera must be returned to the factory for repair if a malfunction occurs.

The lens should only be cleaned using dust free compressed air.

Clean the exterior of the camera with a soft, dry, lint-free cloth. For stubborn dirt, the cloth may be dampened with a mild soap solution.

19.2 Technical Support

Contact Princeton Instruments to find a qualified service center:

Princeton Instruments, Inc.
3660 Quakerbridge Road
Trenton, NJ 08619

Toll Free in USA: +1-877-474-2286
Telephone: +1-609-587-9797
Email: **moreinfo@princetoninstruments.com**
Website: **www.princetoninstruments.com**

19.3 How to Create a Diagnostic Report

1. To record current settings and email a report to your Princeton Instruments technical representative, go to the toolbar and select Tools > Diagnostic Report.
2. Fill out the Diagnostic Report.
3. Select a location to save the file using the Browse button and click Save.
4. Email the file to your Princeton Instruments technical representative.

19.4 Warranty

New Product Warranty

For a period of one (1) year from the date of shipment, Princeton Instruments, Inc. (hereafter referred to as Princeton Instruments), warrants that the imager, and accessories manufactured by Princeton Instruments (collectively the "Product") are in conformity with published specifications and that such items are of good material and workmanship. If any item is defective in material or workmanship or otherwise fails to meet the specifications, or fails to function properly, the Purchaser shall have the right to return such defective or nonconforming products to Princeton Instruments for correction or replacement.

Princeton Instruments agrees to repair or replace, at Princeton Instruments' discretion, without charge any item that is returned to Princeton Instruments for inspection, provided such inspection discloses to the satisfaction of Princeton Instruments that the defects are as specified and conform to the provisions of the New Product Warranty. Princeton Instruments shall have no obligation under this New Product Warranty to provide local repair or replacement services for the Product, but will, at Princeton Instruments' sole discretion, provide repair or replacement services at its own factory or a designated service facility. Products shall not be returned to the Princeton Instruments factory or a designated service facility for inspection, replacement, or repair without specific written authorization from Princeton Instruments. Princeton Instruments will grant such authorization with the issuance of a Return Material Authorization (RMA) number provided that the Purchaser shall have notified Princeton Instruments of any defect or nonconformance within thirty (30) days after Princeton Instruments' shipment of the Product.

THIS WARRANTY DOES NOT APPLY TO THE FOLLOWING CONDITIONS:

Damage caused by a failure to operate the Product in accordance with Princeton Instruments' written instructions as provided in the Princeton Instruments user manual, including but not limited to, environmental specifications; evidence of product being subjected to accidental damage, misuse, abuse, or tampering including the removal, alteration or defacing of Product identification markings; damage resulting from the unauthorized attempt to repair or modify the Product by non- Princeton Instruments personnel; damage caused during shipment.

This warranty and Princeton Instruments' obligation hereunder are in lieu of all other warranties and Princeton Instruments makes no other warranties, express or implied, including, but not limited to, warranties of fitness, non-infringement, or merchantability. Under no circumstances shall Princeton Instruments be liable for special or consequential damages, including, but not limited to, any claimed loss of profits. Princeton Instruments' liability shall be exclusively limited to the repair or replacement of any defective or nonconforming Product and the Purchaser expressly waives any other remedy or measure of damage, statutory or otherwise.

20. Troubleshooting

20.1 Issues with NI-IMAQ Drivers

Issues: Camera shows up in Measurement and Automation Explorer with a yellow exclamation mark next to it.

Camera is listed as a Generic 1394 Desktop Camera under Devices and Interfaces.

Reason: A different driver has been associated with the camera.

Solution: (4-camera controllers only) To change the associated driver, right-click on the camera and select Driver > NI-IMAQ for IEEE 1394a Digital Cameras.

Issue: Camera does not show up in Measurement and Automation Explorer

Reason: (4-camera controllers only) There is a conflict with another driver or you are using a version of NI-IMAQ for IEEE 1394a Cameras earlier than 1.1

Solution: Update the driver from the computer's Device Manager.

1. From the Windows taskbar select Start > Settings > Control Panel.
2. Double-click the System folder.
3. Select the Hardware tab, and click the Device Manager button.
4. Verify that the camera is listed in the Device Manager.
5. Right-click the camera listing and select Properties from the pop-up menu.
6. Click the Update Driver button.
7. Select Install from a list or specific location (Advanced).
8. Select Don't search, I will choose the driver to install. The Hardware Update Wizard appears.
9. Select the NI-IMAQ IEEE 1394a Digital Camera and follow the instructions in the Hardware Update Wizard.
10. Open MAX and verify that your camera appears in the configuration tree.

If your camera still does not show up in the Device Manager, check the following:

- Make sure that you have the latest Service Pack of your respective OS.
- Try installing the camera on another computer.
- Verify that the camera is functioning properly.

20.2 Issues with GigE Vision

Issue: The camera seems to be taking too long to connect but does finally connect.

Check the IP address: if using a static IP, confirm that it is as indicated in Step 4 "Preparing for Image Acquisition." Controller has not finished initializing (about 1 minute). It takes about 1 minute for DHCP to timeout, and then it takes another 30 seconds.

Issue: Camera does not appear in MAX**.**

- Check the software firewall. If active, turn it off.
- Check the IP address: if using a static IP, confirm that it is as indicated in Step 4 "Preparing for Image Acquisition."
- Controller has not finished initializing (about 1 minute). It takes about 1 minute for DHCP to timeout, and it takes another 30 seconds.

Issue: Camera appears but Snap or Grab is not working.

- High Performance driver is not installed.
- Jumbo packets are not selected or are not supported.

Issue: Application reports a Timeout Error.

The software is waiting for an image that it doesn't receive within the timeout period. Check the packet size in the application software that it is compatible with the setting for the network adapter (see Step 3 of acquire Image Data for the First Time). Camera is in Trigger mode (didn't arrive or trigger interval is longer than the timeout value). Increase the trigger frequency or increase the timeout setting.

Issue: Image capture problems (lines or missing data).

Data rate from the camera is higher than the computer is able to support. Use an Intel Pro 1000, jumbo packets, check system resources, and use a direct Ethernet connection between the camera controller and the computer (remove intervening switches/routers).

Issue: Red X appears in MAX.

Another application is already using the camera. MAX and MegaPlus cannot be running at the same time. Only one of these applications can be active at a time. Close MegaPlus and MAX. Restart one of the applications.

Issue: Attribute out of Range or unable get and/or set an attribute.

1. From MAX, go to Camera Attributes.
2. Select UserSets at the bottom.
3. Change UserSetSelector to Default
4. Select UserSetLoad
5. Click on EXECUTE
6. Wait for the camera to disappear from MAX (goes away due to a bug)
7. Close MAX
8. Delete the NI-IMAQdx Data directory. This directory is located in C:\Documents and Settings\All Users\Documents\National Instruments\NI-IMAQdx (for Windows XP/2000) or C:\Users\Public\Public Documents\National Instruments\NI-IMAQdx (for Windows Vista 32-bit)
9. Open MAX.
10. Wait for the camera to re-appear.
11. The problem should be resolved.

Issue: Connection with the camera will be lost after executing a UserSet save and/or a UserSetLoad in MAX.

See instructions for **Attribute out of Range…** If following those instructions does not restore connection with the camera, contact customer support.

21. Bit Windowing Overview

The bit window feature in *MEGAPLUS* cameras allow selecting of which bits (8 bit or 10bit) are output from the 12 bit range digitization of the image. The bit window is set using the bit window controls that are located on the Output Data tab of the Controller Control panel.

Below is an example of where 8 bit output is selected.

Various bit windows are selected by determining whether the start bit is 0, 1, 2, 3 or 4. When the start bit of 4 is selected, the full scale output of the analog to digital (A/D) is output as 8 bits. So a count of 4095 in the 12 bit digitization is output as a count of 255. If a start bit of 3 is selected, then ½ of the A/D full scale is output. If a start bit of 2 is selected, then ¼ of the A/D full scale is output, etc.

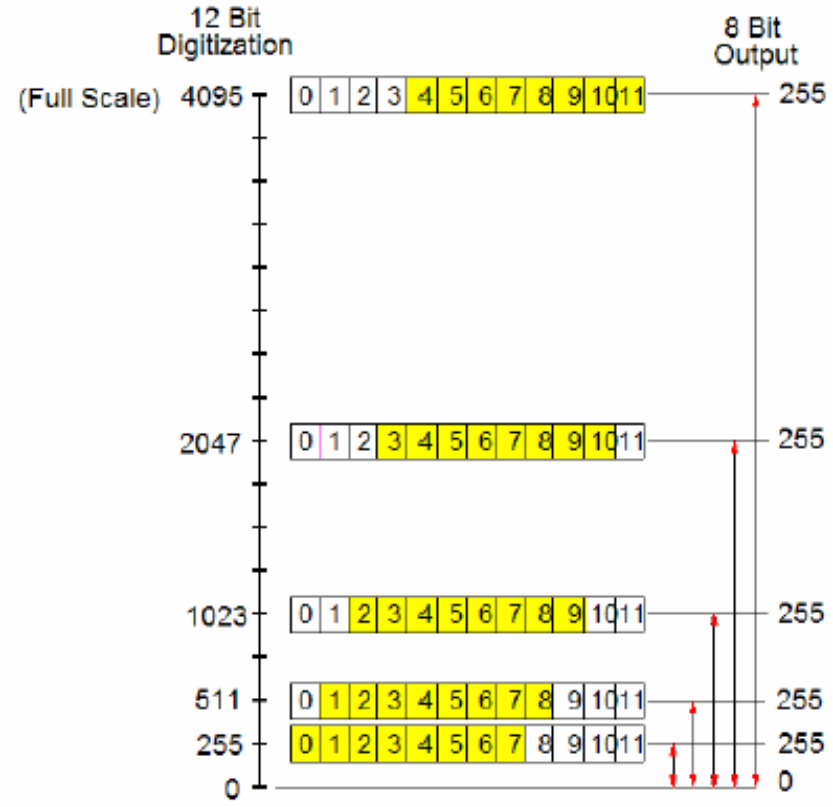


Figure 21.1 Bit Window Example

For more information:

See "How to Set the Bit Windowing" on page 71.

This page intentionally left blank.

22. Color Space Correction

The color Camera Heads in the *MEGAPLUS* product suite provide an RGB output color space that is determined by the spectral properties of the color filter array (CFA) in front of the sensor photosites. While a good quality color image is achieved, it inherently does not conform to a color standard since there are no native standards for color sensors. Instead cameras and other devices are generally mapped to standardized color spaces for accurate color imaging.

Many color spaces are oriented at emulating the visual response of the human eye. CIELab and CIExy are two commonly used color spaces to approximate human perception. For addressing device color space dependence, the Inter*national* Color Consortium has developed a profile system that maps both image input (cameras, scanners, etc.) and image output (printers, monitors, etc.) to a common exchange color space.

The human vision is very non-linear. We can see light over a 6 decade range, which represents an exceptional dynamic range. We do so by perceiving light in a non-linear manner. To match the non-linear visual human perception, the linear image is processed using non-linear multipliers. Commonly this is done by processing image color channels through a Lookup Table (LUT). In addition, since our eyes perceives light as mapped out with the CIE 1931 color, how our brain processes color information differs from how an RGB camera images. Note in the CIE curve shown below how the X component corresponding to red color perception also shows a response in the blue spectral region and how green and red have a substantial overlap. The human brain is able to use this overlapped data to perceive colors that are not perceived with an RGB camera.

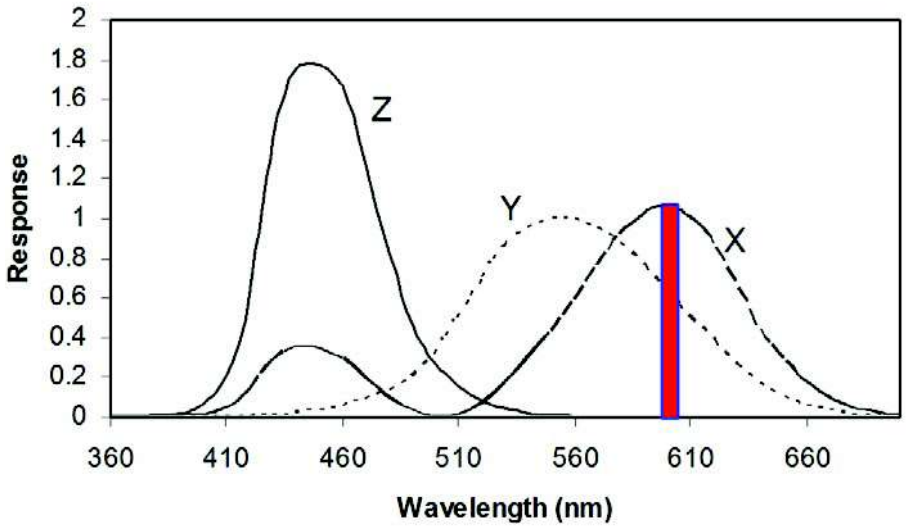


Figure 22.1 CIE 1931 Color Curve

For example, the red spectral band shown below is perceived by both the red and green light receptors in the eye and the brain processes this ratio of information to perceive a color shade. In contrast, the KAI-11002 spectral curve has the same red spectral band shown. Note how the red band is only observed by the red channel. If this red spectral band were at a longer wavelength, the camera response would be unchanged. So the camera sensor and human eye perceive the red spectral band differently.

Color space conversion is a process of using the data from an image measuring device and mathematically approximating a different color space. For example, image information received by the blue image sensor channel might be used to modify the red output data thereby approximating the component of the red visual preceptor that is sensitive to blue light. So this blue image sensor data adjusting the red output is accomplished via one of the matrix components in a 3 x 3 color correction matrix.

The format of the *MEGAPLUS* color correction model is shown below. Lookup tables on the input and output allow mapping from a linear to a non-linear color space. The 3 x 3 color matrix allows conversion from one color space to another.

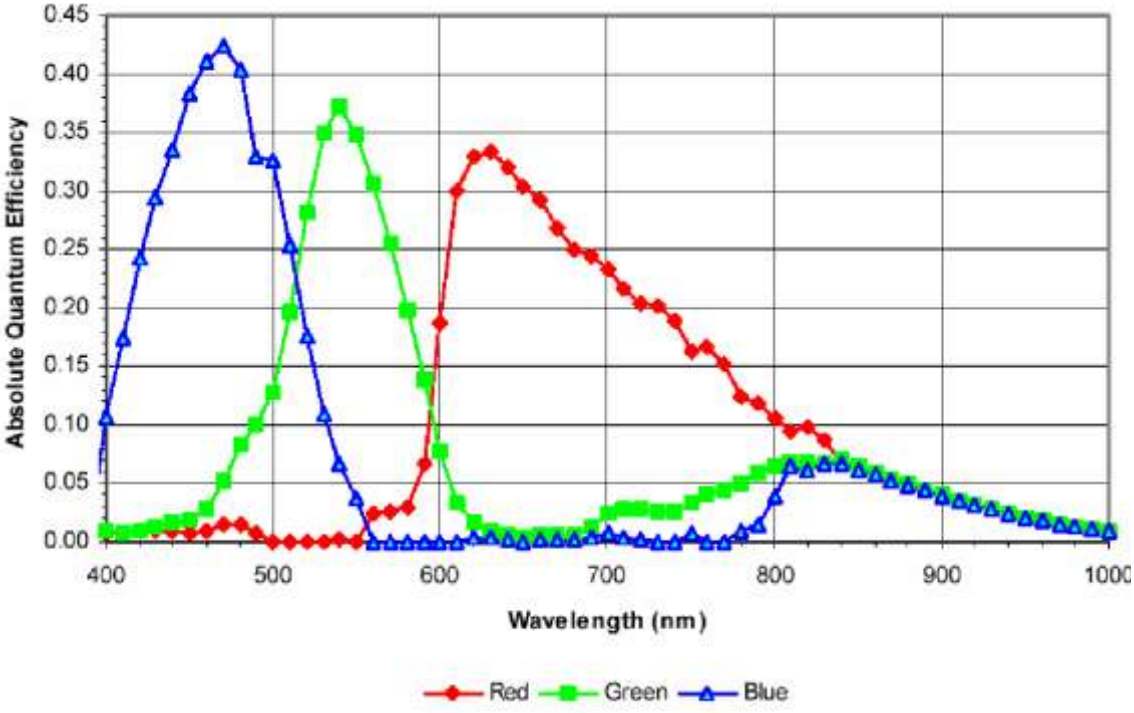


Figure 22.2 KAI-11002 Color Quantum Efficiency

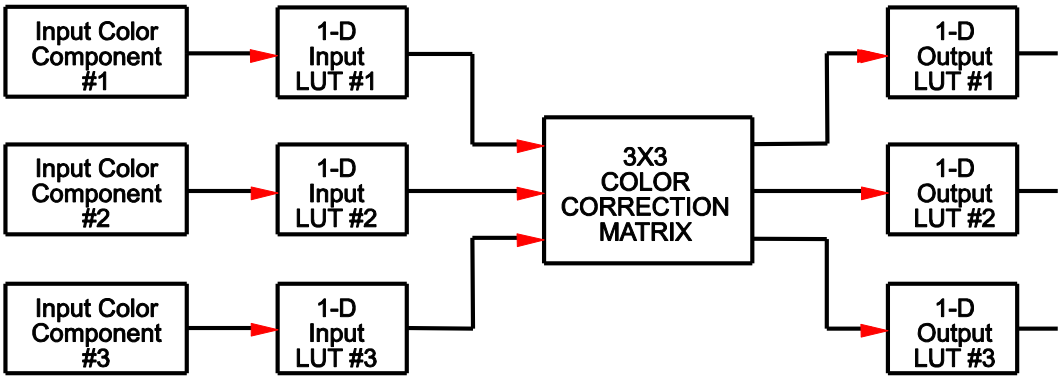

Input Color Component #1
Input Color Component #2
Input Color Component #3
1-D Input LUT #1
1-D Input LUT #2
1-D Input LUT #3
3X3 COLOR CORRECTION MATRIX
1-D Output LUT #1
1-D Output LUT #2
1-D Output LUT #3

Figure 22.3 *MEGAPLUS* Color Correction Model

23. Updating *MEGAPLUS* Camera Controller Firmware via FTP

The *MEGAPLUS* Camera Controller provides a majority of the functionality of the camera via programmable devices such as the controller's processor and main field programmable gate array (FPGA). The functionality of the camera can be changed by modifying the programs in these devices. Therefore, new features and system modifications can be implemented without any changes to the hardware. The processor programs, FPGA configurations, and associated data tables used in the camera are collectively referred to as the camera's "firmware."

The camera's firmware can be updated by downloading new versions of the various system files to the camera controller via an Ethernet connection. This transfer is accomplished using the FTP protocol. The following procedure provides step by step instructions for performing an update. The figures shown in this document are from a Windows 2000 platform. The screen displays on other Windows platforms will be similar, but slightly different depending on the operating system.

Important Notice:
Disable all software firewall programs before updating the firmware.

This includes Windows Firewall, which is automatically installed and started after installing Windows XP Service Pack 2 (SP2). Other software firewall programs include Norton Internet Security, McAfee Security, and Zone Alarm.

Windows Firewall
General
Exceptions
Advanced
Your PC is not protected: turn on Windows Firewall
Windows Firewall helps protect your computer by preventing unauthorized users from gaining access to your computer through the Internet or a network.
On (recommended)
This setting blocks all outside sources from connecting to this computer, with the exception of those selected on the Exceptions tab.
Don't allow exceptions
Select this when you connect to public networks in less secure locations, such as airports. You will not be notified when Windows Firewall blocks programs. Selections on the Exceptions tab will be ignored.
Off (not recommended)
Avoid using this setting. Turning off Windows Firewall may make this computer more vulnerable to viruses and intruders.
Windows Firewall is using your non-domain settings.
What else should I know about Windows Firewall?
OK
Cancel
Make sure Windows Firewall is OFF before updating the firmware. Also check if other software firewall programs are running.
Turn them back on after updating the firmware.

Why?

The firmware update process requires using Windows' built-in FTP service for transferring files to the camera. The transfer of all firmware files is necessary to complete the firmware update. Some or all files may not be transferred, or they may become

corrupt if a software firewall detects the transfer and begins blocking the transfer while it is in progress.

What might happen?
Depending on the computer system and particular software firewall, some or all of the files may be transferred before the software firewall detects a transfer is taking place. If a software firewall detects a transfer that is already in process, then the camera may be left with a mix of old and new files, some of which could be corrupted. If a software firewall blocks the transfer, the file transfer will appear to hang or stop working.

How is it possible to tell if the firmware update completed successfully?
After all the files have been transferred properly, the last line will say "Press any key to continue…". If this line is not seen, then the transfer was interrupted by a software firewall.

What should be done if a software firewall began blocking the transfer before all the files were transferred?
If the update process stalls or if "Press any key to continue…" never appears, then the firmware update process should be restarted before restarting the camera. Before starting the firmware update again (by running updater.bat), ensure that all software firewall programs are disabled or otherwise allow FTP transfers to 192.168.1.2. Using a different computer may be required. Do not power cycle the camera until the firmware update process has completed successfully.

What if PING works ok?
A software firewall can still block the FTP transfer even if PINGing the camera always works. PING is not a good test to see if a software firewall will block the file transfer.

How can I test to see if the FTP will work?
Although it is possible to specially test if FTP ports are being monitored by a software firewall program, it is not possible to provide a simple, fool-proof method that works for all cases. If no software firewall programs have been installed or all software firewall programs have been disabled, then the FTP ports certainly should not be getting monitored for traffic. Please seek assistance from an experienced IT professional if there is any doubt if a software firewall program is running.

23.1 Items Needed for an Update

In order to perform an update, you will need the following:

- A PC running Windows XP, Windows 2000, Windows 98, or Windows 98/ME. (Other platforms can be used if they provide software to perform an FTP transfer.)
- A functional *MEGAPLUS* Camera Controller.
- If connecting point-to-point between the PC and camera, use a CAT 5 crossover patch cable.
- If connecting to the camera via a network hub use a standard CAT 5 network cable connection.

23.2 Overview

If you are already familiar with the update process, the following gives a quick summary of the steps in the update process.

1. Connect the network cable to the Ethernet port on the camera controller and turn on the power.
2. Write down your computer's current IP address and settings.

3. Change your computer's IP address to 192.168.1.1 and the Subnet Mask to 255.255.255.0.
4. Open a DOS window and type "ping" to verify connection with the camera.
5. Exit the DOS window. Locate the update files. If they are in a ZIP folder, extract the update files to a directory on your system.
6. Find the "updater.bat" folder and double-click on **updater.bat** to start the update process.
7. When the update is complete, reset your computer's IP address settings to the original values.

23.3 Performing the Update

You should have received the update via E-mail or on a CD from Princeton Instruments Support. For the purposes of this discussion, it will be assumed that all update files are on a CD. Please adjust the directions as needed for updating from a different location.

WARNING: Downloading files other than those specifically provided to you from Princeton Instruments is not supported and may cause the camera to no longer operate. Download only those files specifically provided to you by Princeton Instruments as an update.

23.3.1 How to Configure the Computer's IP Address

The *MEGAPLUS* Camera Controller is configured for IP Address 192.168.1.2. For the update process, you will need to configure your host computer for IP address 192.168.1.1. In many network configurations today, computers are often configured for automatic assignment of an IP address under DHCP. To perform the update, we will temporarily disable this option and configure the PC for the required update IP address.

1. Find the Network and Dial-up Connections Window on your host computer. This is often accessed via My Computer on the Desktop.
2. Right-click on the icon or listing for your Local Area Network connection to display the pop-up menu for that connection.
3. Select Properties. The Local Area Connection Properties dialog will be displayed. Your system must have the TCP/IP protocol installed in order to perform the update process.
4. Select Internet Protocol (TCI/IP) from the connections list and click on the Properties button.

   NOTE: Before making any changes, note what your current settings are and write them down. You will want to restore these settings after the update is complete.

5. In the Internet Protocol (TCP/IP) Properties dialog of the General Tab, click on the Use the Following IP Address button. Type in the following values:
   - IP Address: 192.168.1.1
   - Subnet Mask: 255.255.255.0
6. Click on the OK button to apply these changes. Some machines may require that you reboot the computer before these changes take effect.

23.3.2 How to Connect to the Camera

Now that the IP address has been set, make sure that the computer is connected to the Camera Controller's Ethernet port via the CAT 5 network cable. Use a Crossover Cable if you are connecting the computer to the Camera Controller (for more information see "Ethernet Cross-over Cable Pin Outs" on page 160).

1. Power up the camera and wait until the Power On LED near the power switch on the rear panel begins a slow and steady blink. This indicates that the system has booted successfully and is ready for the update.
2. Open a DOS shell window. On some systems, there will be an MS-DOS icon that will open the window. If you do not see such an icon on the desktop or listed in the Programs menu, you can also open a DOS window by using the Start Menu and selecting Run.
3. In the Run dialog box, type "cmd" in the Open entry and press the OK button.
4. To verify that you have a good connection to the camera, type in the command:
   "ping 192.168.1.2"

   You should see a response similar to that shown in Figure 23.1 as the computer sends test packets to the camera controller and waits for a response.
5. Type "quit" or "exit" to exit the DOS window.

WARNING: If you do not see a response similar to the one shown, you may not have a good connection to the camera. If you do not have a good connection, DO NOT PROCEED WITH THE UPDATE. Contact Princeton Instruments support for assistance.

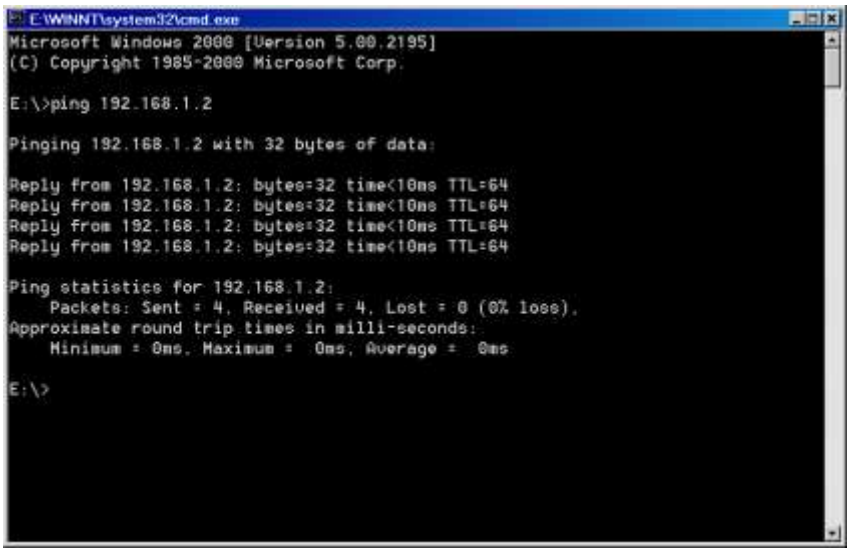


Figure 23.1 Confirming the Connection using "Ping"

23.3.3 How to Perform the Update

1. When communications with the camera has been verified, return to Windows and locate the folder on the CD. Make sure that any update files that are compressed into a ZIP file are extracted into a directory.
2. Open the Updater folder and double-click on the **updater.bat** file. Another DOS window will open and a series of steps will be executed to perform the update. The update process may take several minutes to complete.
3. When the process is complete, the DOS window should close itself. If the window remains open, follow the prompt's instruction or you can type "quit" to exit the window.

NOTE: Do not interrupt the process. A partial update may leave the camera in an inoperable state.

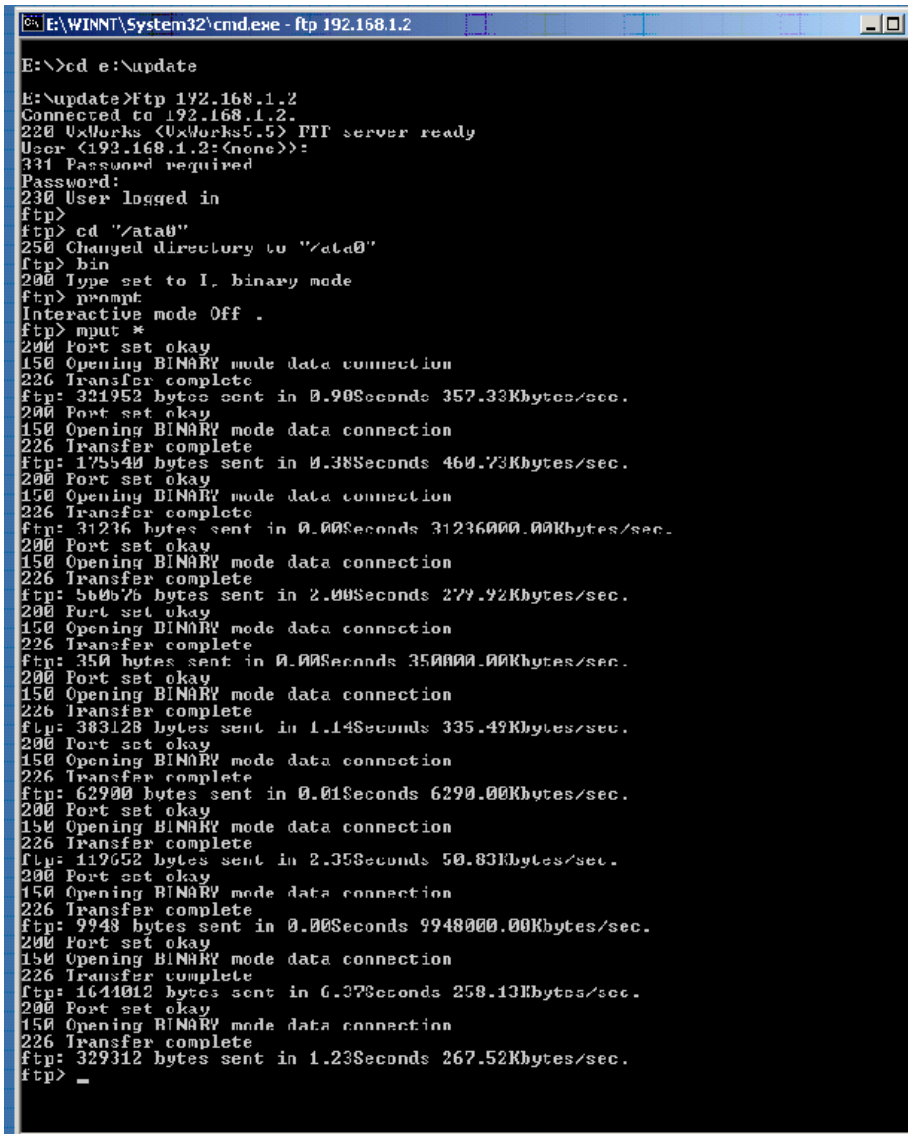

E:\WINNT\System32\cmd.exe - ftp 192.168.1.2
E:\>cd e:\update
E:\update>ftp 192.168.1.2
Connected to 192.168.1.2.
220 VxWorks (VxWorks5.5) FTP server ready
User (192.168.1.2:(none)):
331 Password required
Password:
230 User logged in
ftp>
ftp> cd "/ata0"
250 Changed directory to "/ata0"
ftp> bin
200 Type set to I, binary mode
ftp> prompt
Interactive mode Off .
ftp> mput *
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 321952 bytes sent in 0.90Seconds 357.33Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 175540 bytes sent in 0.38Seconds 460.73Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 31236 bytes sent in 0.00Seconds 31236000.00Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 560676 bytes sent in 2.00Seconds 279.92Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 350 bytes sent in 0.00Seconds 350000.00Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 383128 bytes sent in 1.14Seconds 335.49Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 62900 bytes sent in 0.01Seconds 6290.00Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 119652 bytes sent in 2.35Seconds 50.83Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 9948 bytes sent in 0.00Seconds 9948000.00Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 1641012 bytes sent in 6.37Seconds 258.13Kbytes/sec.
200 Port set okay
150 Opening BINARY mode data connection
226 Transfer complete
ftp: 329312 bytes sent in 1.23Seconds 267.52Kbytes/sec.
ftp> _

Figure 23.2 An Example of an Update Session

23.3.4 How to Reset the IP Address

When the update is complete, return to the Network Connections, TCP/IP Protocol Properties dialog and reset the IP settings to the way they were before the update process.

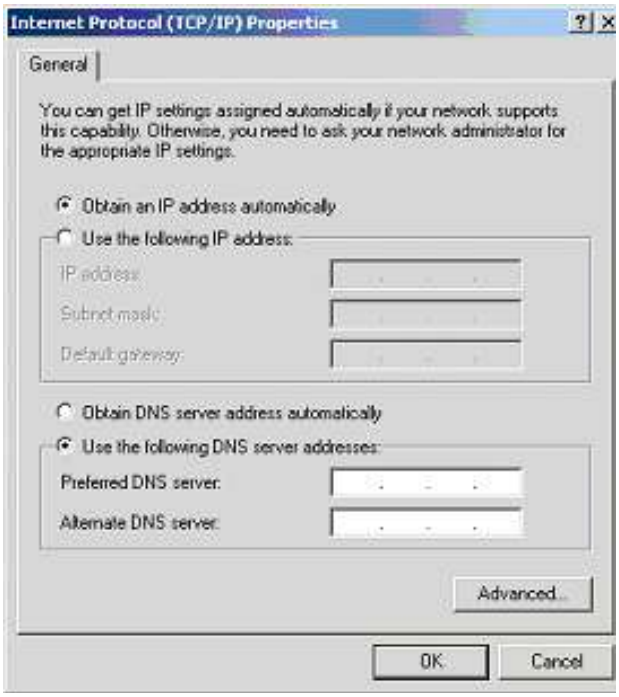
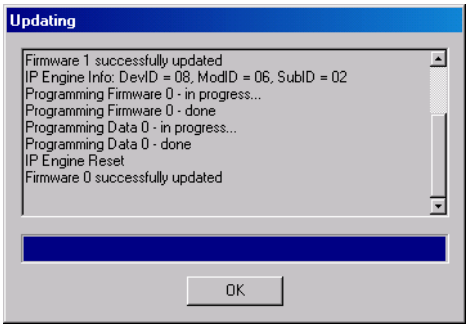

Internet Protocol (TCP/IP) Properties
General
You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.
Obtain an IP address automatically
Use the following IP address:
IP address:
Subnet mask:
Default gateway:
Obtain DNS server address automatically
Use the following DNS server addresses:
Preferred DNS server:
Alternate DNS server:
Advanced...
OK
Cancel

Figure 23.3 Resetting TCP/IP Properties to Pre-update Values

Appendix A. GigE Vision -- Changing from One Camera to Another

1. Acquire data. This confirms that the camera and controller are working properly.
2. Go to C:\Documents and Settings\All Users\Documents\National Instruments\NI-IMAQdx (for Windows XP/2000) or C:\Users\Public\Public Documents\National Instruments\NI-IMAQdx (for Windows Vista 32-bit).


Updating
Firmware 1 successfully updated
IP Engine Info: DevID = 08, ModID = 06, SubID = 02
Programming Firmware 0 - in progress...
Programming Firmware 0 - done
Programming Data 0 - in progress...
Programming Data 0 - done
IP Engine Reset
Firmware 0 successfully updated
OK

3. Delete the Data directory.
4. Make sure the application software is closed.
5. Locate the camera EXE file that corresponds with your camera model. For example, if you have a monochrome ES11000 camera, locate the ES11000M.exe file.
   - If you access to the internet, go to ftp://ftp.princetoninstruments.com/Public/Software/MegaPlus/GEVfiles and download the file (this is the latest version) to your computer.
   - If you do not have to internet access, browse to the GEVfiles subdirectory on the CD.
6. Run the EXE file that corresponds with your camera model. This updates the GigE flash in the controller.
7. When the update is complete, click **OK**.
8. Turn off the controller, connect the new camera, and turn the controller back on. After the controller finishes initializing, you can start the application software, set parameters, and begin acquiring data with the new camera.

This page intentionally left blank.